

平行因子分析理论及其 在通信和信号处理中的应用

张小飞 刘 旭 王成华 李建峰 许凌云 徐大专 著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书介绍了平行因子分析理论及其在通信和信号处理中的应用。平行因子 (Parallel Factor, PARAFAC) 分解属于多线性代数范畴。平行因子分析也称三线性和多线性分解。一般而言, 矩阵分解 (双线性分解) 不是唯一的, 除非施加约束性条件 (正交性、Vandermonde、Toeplitz 和恒模特性等)。PARAFAC 可以看成三维或高维数据阵的低秩分解, PARAFAC 模型的本质特征就是其唯一性。在合适的条件下, PARAFAC 模型本质上是唯一的。平行因子是一种多维数据处理方法, 它充分利用信号的代数性质和分集特性对接收信号进行处理, 并通过多维数据的拟合得到信号处理中需要的各种信息。近年来, 基于 PARAFAC 的信号处理方法因其良好的性能而备受关注, 并已成为通信信号处理中一种新的研究手段。本书详细介绍 PARAFAC 理论数学基础、 k -秩、可辨识性、PARAFAC 分解算法、PARAFAC 分解的 CRB 分析、自适应 PARAFAC 分解、大规模 PARAFAC 分解、扩展 PARAFAC 模型、平行因子压缩感知框架和 PARAFAC 在通信和信号处理中的应用。

本书适合通信与信息系统、信号和信息处理、微波和电磁场、水声等专业的本科高年级学生和研究生阅读。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究。

图书在版编目 (CIP) 数据

平行因子分析理论及其在通信和信号处理中的应用 / 张小飞等著. —北京: 电子工业出版社, 2014.8

ISBN 978-7-121-23735-5

I. ①平… II. ①张… III. ①多线性代数—应用—通信 ②多线性代数—应用—信号处理 IV. ①TN91

中国版本图书馆 CIP 数据核字 (2014) 第 146703 号

责任编辑: 董亚峰 特约编辑: 王 纲

印 刷:

装 订:

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 14.75 字数: 366 千字

版 次: 2014 年 8 月第 1 版

印 次: 2014 年 8 月第 1 次印刷

定 价: 39.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

本书介绍了平行因子分析理论及其在通信和信号处理中的应用。平行因子 (Parallel Factor, PARAFAC) 分解属于多线性代数范畴。一般而言, 矩阵分解 (双线性分解) 不是唯一的, 除非施加约束性条件 (正交性、Vandermonde、Toeplitz 和恒模特性等)。PARAFAC 可以看成三维或高维数据阵的低秩分解, PARAFAC 模型的本质特征就是其唯一性。在合适的条件下, PARAFAC 模型本质上是唯一的。平行因子是一种多维数据处理方法, 它充分利用信号的代数性质和分集特性对接收信号进行处理, 并通过多维数据的拟合得到信号处理中需要的各种信息。近年来, 基于 PARAFAC 的信号处理方法因其良好的性能而备受关注, 并已成为通信信号处理中一种新的研究手段。本书详细介绍 PARAFAC 理论数学基础、 k -秩、可辨识性、PARAFAC 分解算法、PARAFAC 分解的 CRB 分析、自适应 PARAFAC 分解、PARAFAC 模型的扩展和 PARAFAC 在通信和信号处理中的应用。

从 2003 年本书编写小组开展平行因子分析理论及其在通信和信号处理中应用的研究, 历经了 10 多年。本课题组在国内较早研究平行因子分解理论及其在通信和信号处理中的应用。此方面研究得到国家自然科学基金、教育部博士点基金、江苏省博士后基金、中国博士后基金和重点实验室开放课题资助, 培养了 3 名博士生和 11 名硕士生。在平行因子分析理论及其在通信和信号处理中应用方面发表论文 50 多篇, 其中 SCI 检索 30 多篇。平行因子理论有一定突破, 将平行因子方法成功应用于 CDMA 系统的盲信号检测、OFDM 系统中信号检测和参数估计、二维扩频系统的盲信号检测、MC-CDMA 系统的多用户检测、阵列参数估计、极化敏感阵列信号处理、声矢量传感器阵列和 MIMO 雷达参数估计等。

本书从 2011 年开始动笔, 2014 年完成, 写作历经了 3 年。该书在编写过程中, 参考了大量的著作和论文, 在此表示感谢。本书得到了厦门大学水声通信与海洋信息技术教育部重点实验室开放课题项目资助, 在此表示感谢。

本书由张小飞教授、刘旭副教授、王成华教授、李建峰博士、徐大专教授和许凌云博士执笔。刘旭副教授编写了 3.1~3.5、4.3~4.5 节, 徐大专教授和许凌云博士编写了第 5 章

部分内容，其他内容由张小飞教授、王成华教授和李建峰博士完成。在该书写作过程中我们参考了大量学术论文，在此对论文作者表示感谢。在本书编写过程中，还得到了冯宝、王大元、余俊、是莺、冯高鹏、孙中伟、陈未央、吴海浪、陈晨、黄殷杰、王方秋、陈翰、杨刚、曹仁政、余骅欣、蒋驰、周明、张立岑、李书等历届硕士研究生和博士研究生的帮助。

本书作者感谢国家自然科学基金项目（项目号：61371169，61301108）支持。

由于时间仓促，水平有限，加上这一领域仍然处于迅速发展之中，书中不当之处在所难免，敬请读者批评指正。

作 者

2014 年 5 月

符号说明

CMA	恒模算法
CDMA	码分多址
CRB	克拉美-罗界
CS	压缩感知
DOA	波达方向
ESPRIT	借助旋转不变性的信号参数估计
EVD	特征值分解
JADE	联合对角化
LS	最小二乘
LMS	最小均方
MMSE	最小均方误差
MUSIC	多重信号分类
ML	最大似然
MIMO	多输入多输出
OFDM	正交频分复用
PCA	主分量分析
PM	传播算子
PARAFAC	平行因子
PARALIND	基于线性相关平行因子模型
RLS	递归最小二乘
RMSE	求根均方误差
SDMA	空分多址
SNR	信噪比

续表

SVD	奇异值分解
TALS	三线性交替最小二乘
ULA	均匀线阵
UCA	均匀圆阵
Real()	取实部
Imag()	取虚部
diag(.)	表示对角化
(·) ^T	表示转置
(·) ^H	表示共轭转置
(·) [*]	表示复共轭
(·) ⁺	表示广义逆
(·) ⁻¹	表示逆
$\ \cdot \ _F$	Frobenius 范数
angle (·)	取相位
$A \otimes B$	为 Kronecker 积
$A \odot B$	为 Khatri–Rao 积
$A \oplus B$	为 Hadamard 积
$a \circ b$	外积
vec(·)	向量化算子
$E[\cdot]$	求数学期望
tr(·)	矩阵的迹
det(·)	矩阵的行列式
$D_m(\cdot)$	由矩阵的 m 行构造的一个对角矩阵
$\mathbf{1}_{N \times M}$	表示 $N \times M$ 全 1 的矩阵
I_K	表示 $K \times K$ 的单位阵

目 录

第 1 章 绪论	1
1.1 多维矩阵低秩分解	1
1.2 平行因子模型研究现状	2
1.2.1 平行因子模型在通信和信号处理中的应用	2
1.2.2 PARAFAC 分解算法改进	3
1.2.3 PARAFAC 模型的扩展	3
1.2.4 本课题组的工作	4
1.3 本书的安排	4
参考文献	5
第 2 章 数学基础	13
2.1 矩阵代数的相关知识	13
2.1.1 特征值与特征向量	13
2.1.2 广义特征值与广义特征向量	13
2.1.3 矩阵的奇异值分解	14
2.1.4 Toeplitz 矩阵	14
2.1.5 Hankel 矩阵	15
2.1.6 Vandermonde 矩阵	15
2.1.7 Hermitian 矩阵	15
2.1.8 Kronecker 积	16
2.1.9 Khatri-Rao 积	17
2.1.10 Hadamard 积	17
2.1.11 向量化	18
2.1.12 外积	18
2.2 张量代数基础	19
2.2.1 张量代数定义和表示	19
2.2.2 张量的特殊形式	22
2.3 PARAFAC 模型	23
2.3.1 PARAFAC 模型表示	23
2.3.2 PARAFAC 模型的其他表示形式	24
2.4 PARAFAC 分解唯一性	27

2.4.1 矩阵本质相等	27
2.4.2 二维矩阵低秩分解不唯一性	29
2.4.3 PARAFAC 分解唯一性	29
2.5 本章小结	30
参考文献	30
第3章 PARAFAC 基本理论	31
3.1 PARAFAC 模型	31
3.1.1 三线性模型	31
3.1.2 四线性模型或多线性模型	32
3.2 k -秩	33
3.3 可辨识性	34
3.4 PARAFAC 分解	35
3.4.1 三线性交替最小二乘	35
3.4.2 平行因子的快速算法	36
3.4.3 四线性分解和四线性交替最小二乘	39
3.4.4 基于正交约束 PARAFAC 分解	41
3.4.5 结构约束 PARAFAC 分解	43
3.5 PARAFAC 分解的 CRB 分析	50
3.5.1 三线性分解的 CRB 求解	51
3.5.2 约束 CRB 的求解算法	55
3.5.3 “首行已知”约束下三线性分解的 CRB 求解	55
3.5.4 恒模约束下三线性分解的 CRB 求解	57
3.5.5 有限字符约束下三线性分解的 CRB 求解	58
3.5.6 四线性分解的 CRB 求解	59
3.6 自适应 PARAFAC 分解	62
3.6.1 多线性代数基础	62
3.6.2 问题阐述	63
3.6.3 基本思想简介	64
3.6.4 窗的选取	66
3.6.5 PARAFAC-SDT 算法	67
3.6.6 PARAFAC-RLST 算法	71
3.6.7 初始化	74
3.7 大规模 PARAFAC 分解	75
3.7.1 张量符号与基本模型	75
3.7.2 动态张量分解	77
3.7.3 网格 PARAFAC	80
3.8 本章小结	83
参考文献	83

第4章 扩展 PARAFAC 模型	86
4.1 PARALIND 模型	86
4.1.1 PARALIND 模型和分解	86
4.1.2 PARALIND 模型的唯一性	87
4.2 块状 PARAFAC	88
4.2.1 块状 PARAFAC 模型	88
4.2.2 块状 PARAFAC 分解	90
4.3 PARAFAC2	91
4.3.1 PARAFAC2 模型	91
4.3.2 PARAFAC2 分解	92
4.4 PARATUCK2	92
4.4.1 PARATUCK2 模型	92
4.4.2 PARATUCK2 分解	93
4.5 TUCKER	93
4.5.1 TUCKER 模型	93
4.5.2 TUCKER 分解	95
4.6 本章小结	95
参考文献	95
第5章 PARAFAC 压缩感知模型	98
5.1 压缩感知基本原理	98
5.1.1 压缩感知的理论框架	99
5.1.2 矩阵秩最小化理论	101
5.2 PARAFAC 压缩感知理论	102
5.2.1 张量分解的基础	102
5.2.2 PARAFAC 压缩感知框架	103
5.2.3 平行因子模型填充	108
5.3 本章小结	109
参考文献	109
第6章 三线性分解在通信和信号处理中的应用	112
6.1 多天线 OFDM 系中一种基于三线性分解盲载波频偏估计算法	112
6.1.1 数据模型	112
6.1.2 算法原理	113
6.1.3 仿真结果	116
6.2 基于三线性分解的任意矢量传感器阵的二维波达方向估计	120
6.2.1 数据模型	121
6.2.2 三线性分解	122
6.2.3 可辨识性和唯一性	124

6.2.4	算法原理	124
6.2.5	仿真结果	126
6.2.6	小结	130
6.3	阵列天线 MC-CDMA 系统中基于平行因子技术的盲多用户检测算法	131
6.3.1	数据模型	131
6.3.2	阵列天线 MC-CDMA 系统中的盲多用户检测算法	132
6.3.3	仿真结果	134
6.4	单基地 MIMO 雷达中基于自适应 PARAFAC-RLST 的 DOA 跟踪算法	137
6.4.1	数据模型	137
6.4.2	利用自适应 PARAFAC-RLST 进行 DOA 跟踪	137
6.4.3	复杂度分析	140
6.4.4	仿真结果	140
6.5	基于非圆 PARAFAC 任意声矢量阵列下 2D-DOA 估计	143
6.5.1	数据模型	143
6.5.2	基于 NC-PARAFAC 的 2D-DOA 估计算法	144
6.5.3	CRB	148
6.5.4	仿真结果	152
	参考文献	155

第 7 章 四线性分解在通信和信号处理中的应用

7.1	基于四线性分解的均匀面阵的角度和频率联合估计	157
7.1.1	数据模型	157
7.1.2	平行因子四线性模型形成	159
7.1.3	算法描述	160
7.1.4	仿真结果	163
7.2	基于四线性分解的双基地 MIMO 雷达的角度和多普勒频率联合估计	165
7.2.1	双基地 MIMO 雷达时空数据模型	165
7.2.2	基于 PARAFAC 四线性分解的联合估计算法	167
7.2.3	仿真结果	170
	参考文献	173

第 8 章 PARALIND 分解在通信和信号处理中的应用

8.1	非同步 CDMA 系统的 PARALIND 多用户检测	174
8.1.1	数据模型	175
8.1.2	异步 CDMA 系统中基于 PARALIND 的盲空时多用户检测	177
8.1.3	仿真结果	179
8.2	多径下 CDMA 系统的 PARALIND 多用户检测	182
8.2.1	数据模型	182
8.2.2	盲 PARALIND 多用户检测	184
8.2.3	仿真结果	185

8.3	MIMO-OFDM 系统中基于 PARALIND 模型的盲信号检测	188
8.3.1	数据模型	188
8.3.2	基于 PARALIND 的盲符号检测算法	189
8.3.3	仿真结果	191
8.4	声矢量传感器阵列的基于 PARALIND 分解相干二维 DOA 估计算法	194
8.4.1	数据模型	195
8.4.2	相干二维角度估计	195
8.4.3	仿真结果	199
	参考文献	202
第 9 章	PARAFAC 压缩感知理论在通信和信号处理中的应用	204
9.1	基于 PARAFAC 压缩感知模型阵列信号检测	204
9.1.1	数据模型	204
9.1.2	利用三线性模型压缩感知的信号检测算法	205
9.1.3	仿真结果	207
9.2	MIMO 雷达中基于压缩感知平行因子分析的联合角度与多普勒频率估计	208
9.2.1	数据模型	208
9.2.2	联合角度与多普勒频率估计	209
9.2.3	性能分析	214
9.2.4	仿真结果	215
9.3	基于 PARAFAC 填充的面阵 DOA 估计	219
9.3.1	数据模型	219
9.3.2	利用 PARAFAC 填充的 DOA 估计	220
9.3.3	仿真结果	222
	参考文献	224

第1章

绪论

1.1 多维矩阵低秩分解

在线性代数中，“矩阵”是一个常见的概念，它一般具有两个维度，即“行”和“列”，矩阵中的元素可以由行值和列值唯一索引。通常情况下，称传统的矩阵为“二维矩阵”。将矩阵的概念扩展到多维，以三维矩阵 \mathbf{X} 为例，它可以用一个立方体表示，三维矩阵中的元素 x_{ijk} 可以由其“行”、“列”和“高”唯一索引，如图 1.1.1 所示。

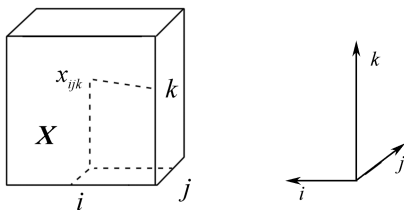


图 1.1.1 三维矩阵示意图

早在 19 世纪 40 年代，Cattell 就对三维矩阵低秩分解进行了研究，并将其列入“因子分析”的范畴。20 世纪 70 年代，Harshman 对多维矩阵低秩分解模型进行了深入研究，他将其命名为著名的平行因子（Parallel Factor, PARAFAC）模型^[1]，Carroll 和 Chang 在同一时间对其进行了研究，并独立地将之称为“规范分解模型”（Canonical Decomposition）^[2]。关于平行因子模型的具体内容将在第 2、3 章进行详细说明。

根据矩阵论的知识可知，在没有任何额外的约束条件时，二维矩阵低秩分解在矩阵的秩大于 1 的情况下是不唯一的。然而，与二维矩阵低秩分解不同，PARAFAC 分解具有分解

唯一特性, J. B. Kruskal 对此做出了重要贡献, 给出了 PARAFAC 模型的分解唯一性定理^[3]。自此, PARAFAC 模型受到了广泛的关注, 众多学者对 PARAFAC 模型的分解唯一性问题进行了研究^[4,5]。PARAFAC 模型的分解唯一性具有重要的应用价值, 通过 PARAFAC 模型的唯一分解可以无模糊地辨识模型参数。到目前为止, PARAFAC 模型作为一种数据分析工具已经广泛地应用于分析化学、心理学、食品学等领域^[6-8]。21 世纪初, PARAFAC 模型被成功地应用于信号处理领域^[9]。从此, 基于 PARAFAC 理论的信号处理技术以其良好的性能而备受关注。本书的后续章节将对 PARAFAC 信号处理技术进行研究, 下面, 简要介绍 PARAFAC 模型在通信和信号处理中的应用。在本书中平行因子分析也称三线性和多线性模型, 或三线性/多线性分解。

1.2 平行因子模型研究现状

1.2.1 平行因子模型在通信和信号处理中的应用

N.D.Sidiropoulos 最早将 PARAFAC 模型引入信号处理领域, 在文献[9]中将直接序列码分多址 (Direct Sequence-Code Division Multiple Access, DS-CDMA) 系统接收信号建模为 PARAFAC 模型, 并设计了新的盲多用户检测算法。该算法可以在信道衰落系数、用户扩频序列均未知的情况下实现多个用户信号的联合检测, 误码率性能仅比非盲的最小均方误差 (Minimum Mean Square Error, MMSE) 多用户检测算法低约 3dB。近年来, 基于 PARAFAC 模型的信号处理思想引起了信号处理领域学者的关注, 研究成果主要集中在以下几方面。

(1) CDMA 系统的盲信号检测。PARAFAC 模型在 CDMA 系统中得到了广泛的应用。文献[10-17]将 PARAFAC 分析方法应用于不同的 CDMA 系统, 解决了多用户检测和参数估计等信号处理问题。基于 PARAFAC 模型的算法在处理 CDMA 信号时, 不需要知道任何用户的扩频码, 它可以实现所有接入用户信号的联合检测。基于上述思想, 文献[18]使用 PARAFAC 分析思想实现了 CDMA 系统小区外的接入用户的盲辨识。文献[19]对基于 PARAFAC 模型的 DS-CDMA 接收机性能进行了详细的仿真和分析。A L.F. Almeida 运用多维矩阵低秩分解的思想对多天线 CDMA 系统进行了建模和分析^[20]。

(2) 阵列信号处理。PARAFAC 分析为阵列天线的波束成形提供了新的思路。N.D.Sidiropoulos 最早将矩形阵列接收信号建模为 PARAFAC 模型, 分别在数据域和子空间域对信号的角度参数进行了估计, 获得了良好的参数估计性能^[21]。近几年, PARAFAC 方法被应用于多种类型的天线阵列中, 如均匀线阵^[22-24]、均匀圆阵^[25,26]、L 形阵列^[27]、极化敏感阵列^[28-30]等。同时, 文献[31]从 PARAFAC 模型分解唯一性的角度分析了均匀线阵波束成形的参数可辨识条件。文献[32-34]给出了一类基于 PARAFAC 模型的近场源定位方法。阵列校准问题同样可以使用 PARAFAC 方法解决^[35]。

(3) 多输入多输出 (Multiple Input Multiple Output, MIMO) 系统。Yuaning Yu 将

PARAFAC 分析方法与高阶累积量相结合, 将 MIMO 信号的三阶/四阶累积量表示成 PARAFAC 模型, 实现了 MIMO 系统的盲信号检测和参数辨识^[36-40]。文献[41-43]使用 PARAFAC 分解对 MIMO 系统的信道进行了均衡和辨识, 得到了良好的效果。PARAFAC 模型的思想同样可以用于设计各种空时编码和空时频编码^[44-46], 接收端可以在信道衰落系数未知的情况下使用 PARAFAC 分解对编码信号进行解码, 同时实现满编码增益, 这是传统的空时编码所无法实现的。

(4) 其他。正交频分复用技术 (Orthogonal Frequency Division Multiplexing, OFDM) 技术是第四代移动通信的关键技术之一, 有学者研究表明, 运用 PARAFAC 分析的思想同样可以实现 OFDM 信号的频偏估计和盲信号检测^[47-49]。此外, PARAFAC 模型还可以解决语音分离^[50]、盲信号分离等问题^[51,52]。

1.2.2 PARAFAC 分解算法改进

在拓展 PARAFAC 分析方法在信号处理领域的应用的同时, 提高算法的性能也是研究的重点之一。PARAFAC 模型的拟合通常使用三线性交替最小二乘算法 (Trilinear Alternative Least Square, TALS) 完成, TALS 是一种交替算法, 其收敛速度和拟合精度直接影响信号处理算法的性能。为了降低算法的运算复杂度, 文献[53]提出了因子压缩方法, 在模型拟合前先将数据在子空间域进行压缩, 然后对压缩后的数据进行拟合, 最后对拟合结果进行处理, 恢复到压缩前的数据空间, 压缩过程降低了算法复杂度。文献[54]根据 TALS 的线性变化性质提出了线性搜索方法 (Linear Search) 以加速 TALS 算法的收敛, 文献[55-56]对其进行可扩展, 提出了增强的线性搜索方法 (Enhanced Linear Search), 并将其成功地用于 DS-CDMA 系统的多用户检测中。有学者指出, 可以通过增加结构约束的方式提高算法的性能, 文献[57,58]将数据的独立性和非负性与 PARAFAC 分析方法相结合, 获得了性能上的提升。针对 PARAFAC 在 CDMA 系统中的应用, 文献[16,59]分别提出了 ITLAS 算法和 OCTALS 算法用于拟合具有 PARAFAC 模型形式的 CDMA 信号, 它们均可以有效地提高模型拟合的收敛速度和拟合精度。传统的基于 PARAFAC 模型的信号处理算法均为“块”算法, 它需要信道在一定时间内状态保持不变, 且复杂度较高。D.Nion 在 2009 年提出了基于 PARAFAC 模型的自适应信号处理算法^[60], 该算法可以跟踪信道状态的改变, 有效地降低了算法的运算复杂度, 为基于 PARAFAC 模型的信号处理技术开辟了新的研究方向。

1.2.3 PARAFAC 模型的扩展

在一些复杂的通信系统中, 虽然系统的接收信号符合三线性模型, 但是它们可能并不严格满足 PARAFAC 模型分解的唯一性条件, 甚至不满足 PARAFAC 模型的形式, 若使用 PARAFAC 模型对这些信号进行分析和处理, 无法得到正确的结果。此时, 需要对 PARAFAC 模型进行扩展。Harshman 在 1974 年提出了 PARAFAC 模型的扩展模型: PARAFAC2 模型^[61],

文献[62,63]对 PARAFAC2 模型进行了深入研究,并将其应用于故障检测和半导体诊断。R Bro 和 N.D.Sidiropoulos 提出了平行线性相关剖面 (Parallel Profile With Linear Dependences, PARALIND) 模型用以处理带有相关剖面的 PARAFAC 模型^[64,65],该模型可以对含有相关剖面的三维数据进行处理。文献[66,67]使用该模型解决了多径信道中的异步 CDMA 信号的多用户检测问题。De Lathauwer 从块状因子分析的角度考虑多维低秩分解问题,提出了基于块的高阶张量分解模型^[68-70],该模型可以应用于多种通信系统^[71-73]。A.L.F. Almeida 博士对基于多维矩阵低秩分解的信号处理算法进行了深入的研究,相继提出了基于块的约束 PARAFAC 模型、约束张量模型和约束 Tucker3 模型对各种复杂的通信系统接收信号进行建模,设计了多种信号处理算法^[20,74-76]。文献[77]将通信信号的结构特征融入 PARAFAC 模型,设计了结构约束 PARAFAC 模型对通信信号进行表征,获得了良好的效果。

1.2.4 本课题组的工作

本课题组在国内较早研究平行因子分解理论及其在通信和信号处理中的应用。此方面研究得到了国家自然科学基金、教育部博士点基金、江苏省博士后科研资助计划项目、中国博士后基金、重点实验室开放课题资助。在此方面培养了两名博士生^[78,79]、11 名硕士生^[80-88]。本课题组在平行因子理论方面有一定突破^[78,79,89,90],并将 PARAFAC 应用到 CDMA 系统的盲信号检测^[12-16,66-67]、OFDM 系统中信号检测和参数估计^[47,84,88,92,95]、二维扩频系统的盲信号检测^[12,96]、多载波码分多址系统^[97]、阵列参数估计^[85,86,26,89,90,98-102]、极化敏感阵列信号处理^[25,29,30,80,83,103]、声矢量传感器阵列^[104-107]和 MIMO 雷达参数估计^[108-112]之中。

1.3 本书的安排

第 2 章介绍矩阵代数、张量代数和 PARAFAC 理论的基础知识,是后续章节的基础。

第 3 章研究 PARAFAC 模型、 k -秩、可辨识性、PARAFAC 分解算法、PARAFAC 分解克拉美-罗界分析 (Cramer-Rao Bound, CRB) 分析、自适应 PARAFAC 分解和大规模 PARAFAC 分解。

第 4 章研究扩展 PARAFAC 模型,如 PARALIND 模型、块状 PARAFAC、PARAFAC2、PARATUCK2 和 TUCKER。

第 5 章研究压缩感知基本原理,以及平行因子压缩感知理论。

第 6 章研究三线性分解理论在通信和信号处理中的应用;将自适应 PARAFAC 应用于 MIMO 雷达系统中,实现单基地 MIMO 雷达角度跟踪。研究一种非圆 (Non-Circular, NC) PARAFAC 算法,该算法可以用于任意声矢量阵列下非圆信号源的 2D-DOA 估计。

第7章就四线性分解理论在通信和信号处理中的应用,给出几个例子。

第8章研究 PARALIND 理论在通信和信号处理中的应用。

第9章研究 PARAFAC 压缩感知理论在通信和信号处理中的应用。

参考文献

- [1] Harshman R A. Foundations of the PARAFAC procedure: models and conditions for an “explanatory” multimodal factor analysis[J]. UCLA Working Papers in Phonetics, 1970, 16: 1-84.
- [2] Carroll J D, Chang J. Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition[J]. Psychometrika, 1970, 35 (3) : 283-319.
- [3] Kruskal J B. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics[J]. Linear Algebra and its Applications, 1977, 18 (2) : 95-138.
- [4] Jiang T, Sidiropoulos N D. Kruskal’s permutation lemma and the identification of CANDECOMP/PARAFAC and bilinear models with constant modulus constraints[J]. IEEE Transactions on Signal Processing, 2004, 52 (9) : 2625-2636.
- [5] Ten Berge J M, Sidiropoulos N D. On uniqueness in CANDECOMP/PARAFAC[J]. Psychometrika, 2002, 67 (3) : 399-409.
- [6] Do T, McIntyre N S. Application of parallel factor analysis and X-ray photoelectron spectroscopy to the initial stages in oxidation of aluminium[J]. Surface Science, 1999, 433: 136-141.
- [7] Bro R. PARAFAC. Tutorial and applications[J]. Chemometrics and Intelligent Laboratory Systems, 1997, 38 (2) : 149-171.
- [8] Rinnan Å. Application of PARAFAC on spectral data[D]. Copenhagen: Department of Food Science, Food Technology, Royal Veterinary and Agricultural University, PH.D. thesis, 2004.
- [9] Sidiropoulos N D, Giannakis G B, Bro R. Blind PARAFAC receivers for DS-CDMA systems[J]. IEEE Transactions on Signal Processing, 2000, 48 (3) : 810-823.
- [10] Sidiropoulos N D, Dimic G Z. Blind multiuser detection in W-CDMA systems with large delay spread[J]. IEEE Signal Processing Letters, 2001, 8 (3) : 87-89.

- [11] Liu X, Sidiropoulos N D, Swami A. Blind high-resolution localization and tracking of multiple frequency hopped signals[J]. IEEE Transactions on Signal Processing, 2002, 50 (4) : 889-901.
- [12] 张小飞, 徐大专. 二维扩展频谱系统下的 PARAFAC 接收机[J]. 南京邮电大学学报 (自然科学版), 2007, 27 (2) : 5-8.
- [13] Zhang X, Xu D. PARAFAC multiuser detection for SIMO-CDMA system[C]. International Conference on Communications, Circuits and Systems Proceedings, 2006, 2: 744-747.
- [14] Zhang X, Xu D. Blind PARAFAC receiver for space-time block-coded CDMA system[C]. International Conference on Communications, Circuits and Systems Proceedings, 2006, 2: 675-678.
- [15] 张小飞, 赵瑞娜, 徐大专. 大时延扩展 CDMA 信道下空时多用户检测[J]. 南京航空航天大学学报, 2007, 39 (2) : 204-207.
- [16] 张小飞, 徐大专. 多径信道中多用户检测和 DOA 估计联合处理方法[J]. 南京航空航天大学学报, 2005, 37 (6) : 776-781.
- [17] Sidiropoulos N D, Bro R. User separation in DS-CDMA systems with unknown long PN spreading codes[C]. 2nd IEEE Workshop on Signal Processing Advances in Wireless Communications, 1999: 194-197.
- [18] Jiang T, Sidiropoulos N D. Blind identification of out-of-cell users in DS-CDMA[J]. EURASIP Journal on Advances in Signal Processing, 1900, 2004 (9) : 1212-1224.
- [19] Jiang T, Sidiropoulos N D. Blind Identification of Out of Cell Users in DS-CDMA[J]. EURASIP Journal on Applied Signal Processing, 2004, 9: 1-13.
- [20] De Almeida A L, Favier G, Mota J C M. Constrained tensor modeling approach to blind multiple-antenna CDMA schemes[J]. IEEE Transactions on Signal Processing, 2008, 56 (6) : 2417-2428.
- [21] Sidiropoulos N D, Bro R, Giannakis G B. Parallel factor analysis in sensor array processing[J]. IEEE Transactions on Signal Processing, 2000, 48 (8) : 2377-2388.
- [22] Zhang X, Yu J, Feng G, et al. Blind direction of arrival estimation of coherent sources using multi-invariance property[J]. Progress in Electromagnetics Research, 2008, 88: 181-195.
- [23] Zhang X, Wang D, Xu D. Novel blind joint direction of arrival and frequency estimation for uniform linear array[J]. Progress in Electromagnetics Research, 2008, 86: 199-215.
- [24] Junli L, Shuyuan Y, Junying Z. A cumulant-based parameter estimation algorithm for near-field sources[J]. Progress in Natural Science, 2007, 17 (8) : 900-905.
- [25] Zhang X, Shi Y, Xu D. Novel blind joint direction of arrival and polarization estimation for polarization-sensitive uniform circular array[J]. Progress in Electromagnetics Research, 2008, 86: 19-37.

- [26] 张小飞, 徐大专. 均匀圆阵中盲联合角度和时延估计方法[J]. 通信学报, 2006, 27(12): 55-60.
- [27] Zhang X, Gao X, Chen W. Improved blind 2D-direction of arrival estimation with L-shaped array using shift invariance property[J]. Journal of Electromagnetic Waves and Applications, 2009, 23 (5-6) : 593-606.
- [28] Zhang X, Xu D. Deterministic blind beamforming for electromagnetic vector sensor array[J]. Progress in Electromagnetics Research, 2008, 84: 363-377.
- [29] Zhang X, Xu D. Blind PARAFAC signal detection for polarization sensitive array[J]. EURASIP Journal on Applied Signal Processing, 2007, 2007 (1) : 1-7.
- [30] Xiaofei Z, Xizhu H, Dazhuan X. Blind PARAFAC signal detection for uniform circular array with polarization sensitive antennas[J]. Transactions of Nanjing University of Aeronautics & Astronautics, 2006, 23 (4) : 291-296.
- [31] Sidiropoulos N D, Liu X. Identifiability results for blind beamforming in incoherent multipath with small delay spread[J]. IEEE Transactions on Signal Processing, 2001, 49 (1) : 228-236.
- [32] 梁军利, 王诗俊, 高丽, 等. 一种无须参数配对的近场源定位新算法[J]. 电子学报, 2007, 35 (6) : 1122-1127.
- [33] 梁军利, 杨树元, 赵峰, 等. 一种新的基于平行因子分析的近场源定位算法[J]. 系统工程与电子技术, 2007, 29 (1) : 33-36.
- [34] 梁军利, 杨树元, 张军英. 一种信号频率及二维到达角联合估计新方法[J]. 电波科学学报, 2007, 22 (6) : 1008-1013.
- [35] 李双勋, 程翥, 王展, 等. 基于 PARAFAC 与阵列旋转的阵列标校方法[J]. 电子与信息学报, 2008, 30 (3) : 656-659.
- [36] Yu Y, Petropulu A P. PARAFAC-based blind estimation of possibly underdetermined convolutive MIMO systems[J]. IEEE Transactions on Signal Processing, 2008, 56 (1) : 111-124.
- [37] Acar T, Yu Y, Petropulu A P. Blind MIMO system estimation based on PARAFAC decomposition of higher order output tensors[J]. IEEE Transactions on Signal Processing, 2006, 54 (11) : 4156-4168.
- [38] Yu Y, Petropulu A P. Improved single PARAFAC decomposition based blind MIMO system estimation[C]. Proceedings of the Fifth IEEE International Symposium on Signal Processing and Information Technology, 2005: 104-108.
- [39] Yu Y, Petropulu A P. Blind Estimation Of A Class Of Under-Determined Convolutive MIMO Systems Using Parafac Decomposition Of Output Tensors[C]. 40th Annual Conference on Information Sciences and Systems, 2006: 1670-1675.

- [40] Yu Y, Petropulu A P. Reduced Complexity Blind Estimation of Under-Determined Convolutional MIMO Systems[C]. 4th Signal Processing Education Workshop, 2006: 239-244.
- [41] Fernandes C E R, Favier G, Mota J C M. Blind channel identification algorithms based on the PARAFAC decomposition of cumulant tensors: the single and multiuser cases[J]. Signal Processing, 2008, 88 (6) : 1382-1401.
- [42] Kibangou A Y, Favier G. Blind equalization of nonlinear channels using a tensor decomposition with code/space/time diversities[J]. Signal Processing, 2009, 89 (2) : 133-143.
- [43] Fernandes C E R, Favier G, Mota J C M. Tensor-based blind channel identification[C]. IEEE International Conference on Communications, 2007: 2728-2732.
- [44] Sidiropoulos N D, Budampati R S. Khatri-Rao space-time codes[J]. IEEE Transactions on Signal Processing, 2002, 50 (10) : 2396-2407.
- [45] De Almeida A L, Favier G, Cavalcante C C, et al. Tensor-based space-time multiplexing codes for MIMO-OFDM systems with blind detection[C]. IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications, 2006: 1-5.
- [46] De Almeida A L, Favier G, Fernandes C A, et al. A trilinear decomposition approach for Space-Time-Frequency multiple-access wireless systems[C]. IEEE 8th Workshop on Signal Processing Advances in Wireless Communications, 2007: 1-5.
- [47] Zhang X, Feng B, Xu D. Blind joint symbol detection and DOA estimation for OFDM system with antenna array[J]. Wireless Personal Communications, 2008, 46 (3) : 371-383.
- [48] Jiang T, Sidiropoulos N D. A direct blind receiver for SIMO and MIMO OFDM systems subject to unknown frequency offset and multipath[C]. 4th IEEE Workshop on Signal Processing Advances in Wireless Communications, 2003: 358-362.
- [49] Rajih M, Comon P, Slock D. A deterministic blind receiver for MIMO OFDM systems[C]. IEEE 7th Workshop on Signal Processing Advances in Wireless Communications, 2006: 1-5.
- [50] Mokios K N, Sidiropoulos N D, Potamianos A. Blind speech separation using parafac analysis and integer least squares[C]. IEEE International Conference on Acoustics, Speech and Signal Processing, 2006, ICASSP 2006 Proceeding, 2006, 5: V73-V76.
- [51] Li T, Sidiropoulos N D. Blind digital signal separation using successive interference cancellation iterative least squares[J]. IEEE Transactions on Signal Processing, 2000, 48 (11) : 3146-3152.
- [52] Rong Y, Vorobyov S A, Gershman A B, et al. Blind spatial signature estimation via time-varying user power loading and parallel factor analysis[J]. IEEE Transactions on Signal Processing, 2005, 53 (5) : 1697-1710.

- [53] Bro R, Andersson C A. Improving the speed of multiway algorithms: Part II: Compression[J]. *Chemometrics and Intelligent Laboratory Systems*, 1998, 42 (1) : 105-113.
- [54] Bro R. Multi-way analysis in the food industry: models, algorithms, and applications[D]. Amsterdam: University of Amsterdam, PH.D. thesis, 1998.
- [55] Rajih M, Comon P, Harshman R A. Enhanced line search: A novel method to accelerate PARAFAC[J]. *SIAM Journal on Matrix Analysis and Applications*, 2008, 30 (3) : 1128-1147.
- [56] Nion D, De Lathauwer L. An enhanced line search scheme for complex-valued tensor decompositions. Application in DS-CDMA[J]. *Signal Processing*, 2008, 88 (3) : 749-755.
- [57] De Vos M, De Lathauwer L, Van Huffel S. Imposing Independence Constraints in the CP Model[J]. *Independent Component Analysis and Signal Separation*, 2007: 33-40.
- [58] Gao H T, Dai D M, Li T H. A novel trilinear decomposition algorithm: Three-dimension non-negative matrix factorization[J]. *Chinese Chemical Letters*, 2007, 18 (4) : 495-498.
- [59] 刘旭, 许宗泽. 正交约束 PARAFAC 的 DS-CDMA 盲接收机[J]. *应用科学学报*, 2009, 27 (2) : 131-136.
- [60] Nion D, Sidiropoulos N D. Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor[J]. *IEEE Transactions on Signal Processing*, 2009, 57 (6) : 2299-2310.
- [61] Harshman R A. PARAFAC2: Mathematical and technical notes[J]. *UCLA Working Papers in Phonetics*, 1972, 22: 30-44.
- [62] Kiers H A, Ten Berge J M, Bro R. PARAFAC2-Part I. A direct fitting algorithm for the PARAFAC2 model[J]. *Journal of Chemometrics*, 1999, 13 (3-4) : 275-294.
- [63] Bro R, Andersson C A, Kiers H A. PARAFAC2—Part II. Modeling chromatographic data with retention time shifts[J]. *Journal of Chemometrics*, 1999, 13 (3-4) : 295-309.
- [64] Bro R, Harshman R A, Sidiropoulos N D, et al. Modeling multi - way data with linearly dependent loadings[J]. *Journal of Chemometrics*, 2009, 23 (7-8) : 324-340.
- [65] Bahram M, Bro R. A novel strategy for solving matrix effect in three-way data using parallel profiles with linear dependencies[J]. *Analytica Chimica Acta*, 2007, 584 (2) : 397-402.
- [66] Zhang X, Feng G, Xu D, et al. Blind PARALIND space-time multiuser detection for asynchronous CDMA system[J]. *Journal of Circuits, Systems, and Computers*, 2009, 18 (3) : 503-517.
- [67] Zhang X, Gao X, Wang Z. Blind PARALIND multiuser detection for smart antenna CDMA system over multipath fading channel[J]. *Progress in Electromagnetics Research*, 2009, 89: 23-38.

- [68] De Lathauwer L. Decompositions of a higher-order tensor in block terms-Part I: Lemmas for partitioned matrices[J]. SIAM Journal on Matrix Analysis and Applications, 2008, 30 (3) : 1022-1032.
- [69] De Lathauwer L. Decompositions of a higher-order tensor in block terms-Part II: Definitions and uniqueness[J]. SIAM Journal on Matrix Analysis and Applications, 2008, 30 (3) : 1033-1066.
- [70] De Lathauwer L, Nion D. Decompositions of a higher-order tensor in block terms-Part III: Alternating least squares algorithms[J]. SIAM Journal on Matrix Analysis and Applications, 2008, 30 (3) : 1067-1083.
- [71] De Lathauwer L, Vandewalle J. Dimensionality reduction in higher-order signal processing and rank- (R_1, R_2, \dots, R_N) reduction in multilinear algebra[J]. Linear Algebra and its Applications, 2004, 391: 31-55.
- [72] Nion D, De Lathauwer L. A block factor analysis based receiver for blind multi-user access in wireless communications[C]. IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings, 2006, 5: V825-V828.
- [73] de Baynast A, Aazhang B, Declercq D, et al. Bayesian blind PARAFAC receivers for DS-CDMA systems[C]. IEEE Workshop on Statistical Signal Processing, 2003: 323-326.
- [74] De Almeida A L, Favier G, Mota J. Constrained tucker-3 model for blind beamforming[J]. Signal Processing, 2009, 89 (6) : 1240-1244.
- [75] De Almeida A L, Favier G, Mota J. Space-time spreading-multiplexing for MIMO wireless communication systems using the PARATUCK-2 tensor model[J]. Signal Processing, 2009, 89 (11) : 2103-2116.
- [76] De Almeida A L, Favier G, Mota J C M, et al. Estimation of frequency-selective block-fading MIMO channels using PARAFAC modeling and alternating least squares[C]. Fortieth Asilomar Conference on Signals, Systems and Computers, 2006: 1630-1634.
- [77] Liu X, Xu Z. Structure Constrained PARAFAC Model with Application to Signal Processing[C]. IEEE International Symposium on Knowledge Acquisition and Modeling Workshop, 2008: 651-654.
- [78] 刘旭. 基于多维矩阵低秩分解的信号处理技术研究[D]. 南京: 南京航空航天大学, 博士, 2009.
- [79] 许凌云. 阵列多参数联合估计算法及应用的研究[D]. 南京: 南京航空航天大学, 博士, 2012.
- [80] 黄希珠. 极化敏感阵列中基于 PARAFAC 的盲信源分离[D]. 南京: 南京航空航天大学, 硕士, 2008.
- [81] 徐锟. TD-SCDMA 系统中的联合检测[D]. 南京: 南京航空航天大学, 硕士, 2008.

- [82] 张胜男. 基于平行因子分析的阵列参数估计[D]. 南京: 南京航空航天大学, 硕士, 2008.
- [83] 是莺. 极化敏感阵列中基于 PARAFAC 的盲 DOA 和极化估计算法[D]. 南京: 南京航空航天大学, 硕士, 2009.
- [84] 冯宝. 阵列天线 OFDM 系统的信号检测与参数估计[D]. 南京: 南京航空航天大学, 硕士, 2009.
- [85] 王大元. 阵列多参数联合估计[D]. 南京: 南京航空航天大学, 硕士, 2010.
- [86] 余俊. 相干信源 DOA 估计[D]. 南京: 南京航空航天大学, 硕士, 2009.
- [87] 陈未央. 声矢量传感器阵列的 DOA 估计[D]. 南京: 南京航空航天大学, 硕士, 2010.
- [88] 冯高鹏. 多天线 OFDM 系统中的频偏估计[D]. 南京: 南京航空航天大学, 硕士, 2012.
- [89] Liu X, Jiang T, Yang L, et al. PARALIND-based identifiability results for parameter estimation via uniform linear array[J]. EURASIP Journal on Advances in Signal Processing, 2012, 2012 (1) : 1-11.
- [90] Liu X, Jiang T, Yang L, et al. Constrained trilinear decomposition with application to array signal processing[J]. Progress in Electromagnetics Research, 2012, 128: 195-214.
- [91] Liu X, Xu Z Z, Lei L. Space-Time Blind Multiuser Detection for Multiuser DS-CDMA and Oversampled Systems[J]. Wireless Personal Communications, 2011, 58 (4) : 759-783.
- [92] Zhang X, Wu H, Li J, et al. Semiblink Channel Estimation and Signal Detection for OFDM System with Receiver Diversity[J]. Wireless Personal Communications, 2012, 66 (1) : 101-115.
- [93] Zhang X, Xu D. Blind CFO estimation algorithm for OFDM systems by using generalized precoding and trilinear model[J]. Journal of Systems Engineering and Electronics, 2012, 23 (1) : 10-15.
- [94] Zhang X, Gao X, Xu D. Novel blind carrier frequency offset estimation for OFDM system with multiple antennas[J]. IEEE Transactions on Wireless Communications, 2010, 9 (3) : 881-885.
- [95] Zhang X, Wang F, Xu D. Blind signal detection algorithm for MIMO-OFDM systems over multipath channel using PARALIND model[J]. IET Communications, 2011, 5 (5) : 606-611.
- [96] Zhang X, Xu D. Blind source separation for two-dimension spread spectrum system based on trilinear decomposition[J]. Journal of Circuits, Systems, and Computers, 2008, 17 (2) : 297-308.
- [97] Zhang X, Feng G, Gao X, et al. Blind multiuser detection for MC-CDMA with antenna array[J]. Computers & Electrical Engineering, 2010, 36 (1) : 160-168.
- [98] Xu L, Zhang X, Xu Z, et al. Joint 2D-DOA and Frequency Estimation for L-Shaped Array Using Iterative Least Squares Method[J]. International Journal of Antennas and

- Propagation, 2012, 2012: 1-8.
- [99] Zhang X, Feng G, Yu J, et al. Angle-frequency estimation using trilinear decomposition of the oversampled output[J]. *Wireless personal communications*, 2009, 51 (2) : 365-373.
- [100] 许凌云, 张小飞, 许宗泽. 平面阵列下的二维角度和频率联合估计[J]. *应用科学学报*, 2011, 29 (2) : 187-193.
- [101] 许凌云, 张小飞, 许宗泽. 基于平行因子四线性分解的二维角度和频率联合估计[J]. *电子与信息学报*, 2011, 33 (8) : 1889-1894.
- [102] Xiaofei Z, Jianfeng L, Lingyun X. Novel two-dimensional DOA estimation with L-shaped array[J]. *EURASIP Journal on Advances in Signal Processing*, 2011, 2011 (1) : 1-7.
- [103] Shi Y, Zhang X. Quadrilinear decomposition-based blind signal detection for polarization sensitive uniform square array[J]. *Progress in Electromagnetics Research*, 2008, 87: 263-278.
- [104] 许凌云, 张小飞, 许宗泽, 等. 基于任意声矢量传感器阵列的角度和频率估计算法[J]. *仪器仪表学报*, 2012, 33 (10) : 2234-2240.
- [105] Zhang X, Li J, Chen H, et al. Trilinear decomposition-based two-dimensional DOA estimation algorithm for arbitrarily spaced acoustic vector-sensor array subjected to unknown locations[J]. *Wireless Personal Communications*, 2012, 67 (4) : 859-877.
- [106] Zhang X, Cao R, Zhou M. Noncircular-PARAFAC for 2D-DOA estimation of noncircular signals in arbitrarily spaced acoustic vector-sensor array subjected to unknown locations[J]. *EURASIP Journal on Advances in Signal Processing*, 2013, 2013 (1) : 1-10.
- [107] Zhang X, Zhou M, Li J. A PARALIND Decomposition-Based Coherent Two-Dimensional Direction of Arrival Estimation Algorithm for Acoustic Vector-Sensor Arrays[J]. *Sensors*, 2013, 13 (4) : 5302-5316.
- [108] Zhang X, Xu Z, Xu L, et al. Trilinear decomposition-based transmit angle and receive angle estimation for multiple-input multiple-output radar[J]. *IET Radar, Sonar & Navigation*, 2011, 5 (6) : 626-631.
- [109] 许凌云, 张小飞, 许宗泽. 基于圆阵双基地 MIMO 雷达多维角度联合估计[J]. *数据采集与处理*, 2012, 27 (3) : 279-283.
- [110] 许凌云, 张小飞, 许宗泽. 双基地 MIMO 雷达二维 DOD 和二维 DOA 联合估计[J]. *应用科学学报*, 2012, 30 (3) : 270-274.
- [111] 李建峰, 张小飞. 基于四线性分解的双基地 MIMO 雷达的角度和多普勒频率联合估计[J]. *航空学报*, 2012, 33 (8) : 1474-1482.
- [112] Wu H, Zhang X, Feng G, et al. DOA Tracking in Monostatic MIMO Radar Using PARAFAC-RLST Algorithm[C]. *The 3rd International Conference on Information Science and Engineering*, 2011: 958-961.

第2章

数学基础

本章将介绍矩阵代数、张量代数和 PARAFAC 理论的基础知识，为后续章节奠定理论基础。

2.1 矩阵代数的相关知识

矩阵代数是本书重要的数学基础，书中很多内容都离不开矩阵代数的相关知识^[1,2]。

2.1.1 特征值与特征向量

令 $A \in C^{n \times n}$ ， $e \in C^n$ ，若标量 λ 和非零向量 e 满足方程

$$Ae = \lambda e, e \neq 0 \quad (2.1.1)$$

则称 λ 是矩阵 A 的特征值， e 是与 λ 对应的特征向量。特征值与特征向量总是成对出现，称 (λ, e) 为矩阵 A 的特征对，特征值可能为零，但是特征向量一定非零。

2.1.2 广义特征值与广义特征向量

令 $A, B \in C^{n \times n}$ ， $e \in C^n$ ，若标量 λ 和非零向量 e 满足方程

$$Ae = \lambda Be, e \neq 0 \quad (2.1.2)$$

则称 λ 是矩阵 A 相对于矩阵 B 的广义特征值， e 是与 λ 对应的广义特征向量。如果矩

阵 \mathbf{B} 非满秩, 那么 λ 可以是任意值 (包括零)。

当矩阵 \mathbf{B} 为单位阵时, 式 (2.1.2) 就成为了普通的特征值问题, 因此式 (2.1.2) 可以看成对普通特征值问题的推广。

2.1.3 矩阵的奇异值分解

对于复矩阵 $\mathbf{A}_{m \times n}$, 称 $\mathbf{A}^H \mathbf{A}$ 的 n 个特征根 λ_i 的算术根 $\sigma_i = \sqrt{\lambda_i} (i=1, 2, \dots, n)$ 为 \mathbf{A} 的奇异值。若记 $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$, 其中 $\sigma_1, \sigma_2, \dots, \sigma_r$ 是 \mathbf{A} 的全部非零奇异值, 则称 $m \times n$ 矩阵

$$\mathbf{S} = \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix} \quad (2.1.3)$$

为 \mathbf{A} 的奇异值矩阵。

奇异值分解定理: 对于 $m \times n$ 维矩阵 \mathbf{A} , 则分别存在一个 $m \times m$ 维酉矩阵 \mathbf{U} 和一个 $n \times n$ 维酉矩阵 \mathbf{V} , 使得

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^H \quad (2.1.4)$$

其中上标 H 表示矩阵的共轭转置。

2.1.4 Toeplitz 矩阵

定义: 具有 $2n-1$ 个元素的 n 阶矩阵

$$\mathbf{A} = \begin{bmatrix} a_0 & a_{-1} & a_{-2} & \cdots & a_{-n+1} \\ a_1 & a_0 & a_{-1} & \cdots & a_{-n+2} \\ a_2 & a_1 & a_0 & \cdots & a_{-n+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n-1} & a_{n-2} & a_{n-3} & \cdots & a_0 \end{bmatrix} \quad (2.1.5)$$

称为 Toeplitz 矩阵, 简称 \mathbf{T} 矩阵。

\mathbf{T} 矩阵也可简记为

$$\mathbf{A} = (a_{-j+i})_1^n \quad (2.1.6)$$

记号 $(a_{-j+i})_1^n$ 中的 1 和 n 表示矩阵 \mathbf{A} 元素的下标, $i, j=1, 2, \dots, n$ 。 \mathbf{T} 矩阵完全由第 1 行和第 1 列的 $2n-1$ 个元素确定。

可见, Toeplitz 矩阵中位于任意一条平行于主对角线的直线上的元素全都是相等的, 且

关于副对角线对称。在数据信息处理过程中,有限元素法、概率统计以及滤波理论等领域,常常遇到具有 $2n-1$ 个元素的 n 阶 Toeplitz 矩阵。

2.1.5 Hankel 矩阵

定义: 具有以下形式的 $n+1$ 阶矩阵

$$\mathbf{H} = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_n \\ a_1 & a_2 & a_3 & \cdots & a_{n+1} \\ a_2 & a_3 & a_4 & \cdots & a_{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_n & a_{n+1} & a_{n+2} & \cdots & a_{2n} \end{bmatrix} \quad (2.1.7)$$

称为 Hankel 矩阵或正交对称矩阵 (Orthosymmetric Matrix)。

可见 Hankel 矩阵完全由其第 1 行和第 n 列的 $2n+1$ 个元素确定。其中沿着所有垂直于主对角线的直线上有相同的元素。

2.1.6 Vandermonde 矩阵

定义: 具有以下形式的 $m \times n$ 阶矩阵

$$\mathbf{V}(a_1, a_2, \cdots, a_n) = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ a_1 & a_2 & a_3 & \cdots & a_n \\ a_1^2 & a_2^2 & a_3^2 & \cdots & a_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1^{m-1} & a_2^{m-1} & a_3^{m-1} & \cdots & a_n^{m-1} \end{bmatrix} \quad (2.1.8)$$

称为 Vandermonde 矩阵。Vandermonde 矩阵 $\mathbf{V}(a_1, a_2, \cdots, a_n)$ 的转置也称 Vandermonde 矩阵。

如果 $a_i \neq a_j$, 那么 $\mathbf{V}(a_1, a_2, \cdots, a_n)$ 是非奇异的。如果 $m \neq n$, 称式(2.1.8)为拟 Vandermonde 矩阵。如果 $m = n$, 称式(2.1.8)为广义 Vandermonde 矩阵。相应的行列式分别称为 Vandermonde 行列式和广义 Vandermonde 行列式。

2.1.7 Hermitian 矩阵

如果矩阵 $\mathbf{A}_{n \times n}$ 满足

$$\mathbf{A} = \mathbf{A}^H \quad (2.1.9)$$

则 \mathbf{A} 称为 Hermitian 矩阵。Hermitian 矩阵包括以下主要性质:

(1) Hermitian 矩阵的所有特征值都是实的。

(2) Hermitian 矩阵对应于不同特征值的特征向量相互正交。

(3) Hermitian 矩阵可分解为 $A = E\Lambda E^H = \sum_{i=1}^n \xi_i \mathbf{e}_i \mathbf{e}_i^H$ 的形式，这一分解称为谱定理，也就是矩阵 A 的特征值分解定理，其中 $\Lambda = \text{diag}(\xi_1, \xi_2, \dots, \xi_n)$ ， $E = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n]$ 是由特征向量构成的酉矩阵。

2.1.8 Kronecker 积

定义： $p \times q$ 矩阵 A 和 $m \times n$ 矩阵 B 的 Kronecker 积记为 $A \otimes B$ ，它是一个 $pm \times qn$ 矩阵，定义为

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1q}B \\ a_{21}B & a_{22}B & \cdots & a_{2q}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1}B & a_{p2}B & \cdots & a_{pq}B \end{bmatrix} \quad (2.1.10)$$

Kronecker 积有一个重要的性质，记： $U \in \mathbb{C}^{m \times n}$ 、 $V \in \mathbb{C}^{n \times p}$ 、 $W \in \mathbb{C}^{p \times q}$ ，以下等式成立：

$$\text{vec}(UVW) = (W^T \otimes U) \text{vec}(V)$$

其中， $\text{vec}(\bullet)$ 为向量化算子，如矩阵 A 为 $I \times R$ 的矩阵， $\text{vec}(A)$ 具有下面的形式：

$$\mathbf{a} = \text{vec}(A) = \begin{bmatrix} a_{1,1} \\ \vdots \\ a_{I,1} \\ \vdots \\ a_{1,R} \\ \vdots \\ a_{I,R} \end{bmatrix} \in \mathbb{C}^{IR \times 1}$$

Kronecker 积具有以下一些性质：

$$A \otimes (aB) = a(A \otimes B)$$

$$(A \otimes B)^T = A^T \otimes B^T$$

$$(A + B) \otimes C = A \otimes C + B \otimes C$$

$$A \otimes (B + C) = A \otimes B + A \otimes C$$

$$A \otimes (B \otimes C) = (A \otimes B) \otimes C$$

$$(A \otimes B)(C \otimes D) = AC \otimes BD$$

$$(A \otimes B)^+ = A^+ \otimes B^+$$

$$\text{vec}(AYB) = (B^T \otimes A) \text{vec}(Y)$$

$$\text{tr}(A \otimes B) = \text{tr}(A) \text{tr}(B)$$

2.1.9 Khatri-Rao 积

考虑两个矩阵 $\mathbf{A} (I \times F)$ 和 $\mathbf{B} (J \times F)$ ，它们的 Khatri-Rao 积 $\mathbf{A} \odot \mathbf{B}$ 为一个 $IJ \times F$ 维矩阵，其定义为：

$$\begin{aligned}\mathbf{A} \odot \mathbf{B} &= [\mathbf{a}_1 \otimes \mathbf{b}_1, \dots, \mathbf{a}_F \otimes \mathbf{b}_F] \\ &= [\text{vec}(\mathbf{b}_1 \mathbf{a}_1^T), \dots, \text{vec}(\mathbf{b}_F \mathbf{a}_F^T)]\end{aligned}\quad (2.1.11)$$

其中， \mathbf{a}_f 为 \mathbf{A} 的第 f 列， \mathbf{b}_f 为 \mathbf{B} 的第 f 列，即 Khatri-Rao 积是列向量的 Kronecker 积。

Khatri-Rao 积具有以下一些性质：

$$\begin{aligned}\mathbf{A} \odot (\mathbf{B} \odot \mathbf{C}) &= (\mathbf{A} \odot \mathbf{B}) \odot \mathbf{C} \\ (\mathbf{A} + \mathbf{B}) \odot \mathbf{C} &= \mathbf{A} \odot \mathbf{C} + \mathbf{B} \odot \mathbf{C} \\ \mathbf{A} \odot \mathbf{B} &\neq \mathbf{B} \odot \mathbf{A} \\ (\mathbf{A} \odot \mathbf{B})^T (\mathbf{A} \odot \mathbf{B}) &= (\mathbf{A}^T \mathbf{A}) \oplus (\mathbf{B}^T \mathbf{B}) \\ (\mathbf{A} \odot \mathbf{B})^+ &= [(\mathbf{A} \odot \mathbf{B})^T (\mathbf{A} \odot \mathbf{B})]^{-1} (\mathbf{A} \odot \mathbf{B})^T = [(\mathbf{A}^T \mathbf{A}) \oplus (\mathbf{B}^T \mathbf{B})]^{-1} (\mathbf{A} \odot \mathbf{B})^T \\ ([\mathbf{A} \odot \mathbf{B}]^T)^+ &= (\mathbf{A} \odot \mathbf{B}) [(\mathbf{A}^T \mathbf{A}) \oplus (\mathbf{B}^T \mathbf{B})]^{-1}\end{aligned}$$

令 $\mathbf{x} \in \mathbb{C}^R$ ，Khatri-Rao 积具有以下性质：

$$\text{unvec}((\mathbf{A} \odot \mathbf{B})\mathbf{x}, J, I) = \mathbf{B} \text{diag}(\mathbf{x})\mathbf{A}^T$$

其中， $\text{unvec}(\bullet)$ 是矩阵化算子，它是 $\text{vec}(\bullet)$ 的逆运算，具有以下形式：

$$\text{unvec}(\mathbf{a}, I, R) = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,R} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,R} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I,1} & a_{I,2} & \cdots & a_{I,R} \end{bmatrix} = \mathbf{A}$$

$\text{diag}(\mathbf{x})$ 表示一个对角矩阵，其元素为向量 \mathbf{x} 中的元素。

2.1.10 Hadamard 积

矩阵 $\mathbf{A} \in \mathbb{C}^{I \times J}$ 和 $\mathbf{B} \in \mathbb{C}^{I \times J}$ 的 Hadamard 积定义为

$$\mathbf{A} \oplus \mathbf{B} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} & \cdots & a_{1J}b_{1J} \\ a_{21}b_{21} & a_{22}b_{22} & \cdots & a_{2J}b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}b_{I1} & a_{I2}b_{I2} & \cdots & a_{IJ}b_{IJ} \end{bmatrix}$$

2.1.11 向量化

通常，张量和矩阵比较方便使用向量来表示，定义矩阵 $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_T] \in \mathbb{R}^{I \times T}$ 的向量化为^[3]

$$\mathbf{y} = \text{vec}(\mathbf{Y}) = [\mathbf{y}_1^T, \mathbf{y}_2^T, \dots, \mathbf{y}_T^T]^T \in \mathbb{R}^{IT} \quad (2.1.12)$$

vec 算子用于将矩阵 \mathbf{Y} 的所有列堆积成一个向量，重塑 (Reshape) 是向量化的逆函数，它将一个向量转化成一个矩阵，例如， $\text{reshape}(\mathbf{y}, I, T) \in \mathbb{R}^{I \times T}$ 可定义为 (使用 MATLAB 表示法并类似于 MATLAB 中的 reshape 函数)：

$$\text{reshape}(\mathbf{y}, I, T) = [\mathbf{y}(1:I), \mathbf{y}(I+1:2I), \dots, \mathbf{y}((T-1)I+1:IT)] \in \mathbb{R}^{I \times T} \quad (2.1.13)$$

类似地，定义张量 $\underline{\mathbf{Y}}$ 的向量化为相应的模-1 展开矩阵 $\mathbf{Y}_{(1)}$ ，例如，三阶张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I \times T \times Q}$ 的向量化可写成以下形式，

$$\text{vec}(\underline{\mathbf{Y}}) = \text{vec}(\mathbf{Y}_{(1)}) = [\text{vec}(\mathbf{Y}_{::1})^T, \text{vec}(\mathbf{Y}_{::2})^T, \dots, \text{vec}(\mathbf{Y}_{::Q})^T]^T \in \mathbb{R}^{ITQ} \quad (2.1.14)$$

vec 算子的基本性质包括：

$$\text{vec}(c \mathbf{A}) = c \text{vec}(\mathbf{A}) \quad (2.1.15)$$

$$\text{vec}(\mathbf{A} + \mathbf{B}) = \text{vec}(\mathbf{A}) + \text{vec}(\mathbf{B}) \quad (2.1.16)$$

$$\text{vec}(\mathbf{A})^T \text{vec}(\mathbf{B}) = \text{tr}(\mathbf{A}^T \mathbf{B}) \quad (2.1.17)$$

$$\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B}) \quad (2.1.18)$$

2.1.12 外积

张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ 和张量 $\underline{\mathbf{X}} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_M}$ 的外积由下式给出：

$$\underline{\mathbf{Z}} = \underline{\mathbf{Y}} \circ \underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N \times J_1 \times J_2 \times \dots \times J_M} \quad (2.1.19)$$

其中

$$z_{i_1, i_2, \dots, i_N, j_1, j_2, \dots, j_M} = y_{i_1, i_2, \dots, i_N} x_{j_1, j_2, \dots, j_M} \quad (2.1.20)$$

注意到，张量 $\underline{\mathbf{Z}}$ 包含 $\underline{\mathbf{Y}}$ 和 $\underline{\mathbf{X}}$ 的元素之间的俩俩之积的所有可能的组合。

特别地，两个向量 $\mathbf{a} \in \mathbb{R}^I$ 和 $\mathbf{b} \in \mathbb{R}^J$ 的外积得到一个秩-1 矩阵

$$\mathbf{A} = \mathbf{a} \circ \mathbf{b} = \mathbf{a} \mathbf{b}^T \in \mathbb{R}^{I \times J} \quad (2.1.21)$$

三个向量 $\mathbf{a} \in \mathbb{R}^I$ 、 $\mathbf{b} \in \mathbb{R}^J$ 和 $\mathbf{c} \in \mathbb{R}^Q$ 的外积得到一个三阶的秩-1 张量：

$$\underline{\mathbf{Z}} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \in \mathbb{R}^{I \times J \times Q} \quad (2.1.22)$$

其中

$$z_{ijq} = a_i b_j c_q \quad (2.1.23)$$

2.2 张量代数基础

2.2.1 张量代数定义和表示

为了计算张量和矩阵乘积，我们需要指定被矩阵的列或行相乘的张量的模（表 2.2.1）^[3]。

定义 2.2.1（张量矩阵的 n 模乘积）：张量 $\underline{G} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$ 和矩阵 $\mathbf{A} \in \mathbb{R}^{I_n \times J_n}$ 的 n 模乘积是一个张量 $\underline{Y} \in \mathbb{R}^{J_1 \times \cdots \times J_{n-1} \times I_n \times J_{n+1} \times \cdots \times J_N}$ ，其元素为^[3]

$$y_{j_1, j_2, \dots, j_{n-1}, i_n, j_{n+1}, \dots, j_N} = \sum_{j_n=1}^{J_n} g_{j_1, j_2, \dots, j_N} a_{i_n, j_n} \quad (2.2.1)$$

张量-矩阵的乘积可用于一些模的级联，且可交换，即

$$(\underline{G} \times_n \mathbf{A}) \times_m \mathbf{B} = (\underline{G} \times_m \mathbf{B}) \times_n \mathbf{A} = \underline{G} \times_n \mathbf{A} \times_m \mathbf{B} \quad (m \neq n) \quad (2.2.2)$$

对于适当维数的矩阵 \mathbf{A} 和矩阵 \mathbf{B} ，其重复的 n 模张量-矩阵乘积可简化为

$$(\underline{G} \times_n \mathbf{A}) \times_n \mathbf{B} = \underline{G} \times_n (\mathbf{B}\mathbf{A}) \quad (2.2.3)$$

对于张量 $\underline{G} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$ 和矩阵集 $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times J_n}$ ，它们所有可能的 n ($n=1, 2, \dots, N$) 模乘积可表示为

$$\underline{G} \times \{\mathbf{A}\} = \underline{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)} \quad (2.2.4)$$

结果张量的维数为 $I_1 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \times \cdots \times I_N$ 。除了一个模的张量乘积可表示为^[3]

$$\underline{G} \times_{-n} \{\mathbf{A}\} = \underline{G} \times_1 \mathbf{A}^{(1)} \cdots \times_{n-1} \mathbf{A}^{(n-1)} \times_{n+1} \mathbf{A}^{(n+1)} \cdots \times_N \mathbf{A}^{(N)} \quad (2.2.5)$$

给定张量的维数为 $I_1 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \times \cdots \times I_N$ 。以上记号取自参考文献[3]。

不难证明这些运算满足以下特性：

$$[\underline{G} \times \{\mathbf{A}\}]_{(n)} = \mathbf{A}^{(n)} \mathbf{G}_{(n)} [\mathbf{A}^{(N)} \otimes \mathbf{A}^{(N-1)} \cdots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \cdots \otimes \mathbf{A}^{(1)}]^T \quad (2.2.6)$$

定义 2.2.2（张量-向量的 n 模乘积）：张量 $\underline{Y} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ 与向量 $\mathbf{a} \in \mathbb{R}^{I_n}$ 的 n 模乘积可表示为^[3]

$$\underline{Y} \bar{\times}_n \mathbf{a} \quad (2.2.7)$$

其维度为 $I_1 \times \cdots \times I_{n-1} \times I_{n+1} \times \cdots \times I_N$ ，即

$$\underline{Z} = \underline{Y} \bar{\times}_n \mathbf{a} \in \mathbb{R}^{I_1 \times \cdots \times I_{n-1} \times I_{n+1} \times \cdots \times I_N} \quad (2.2.8)$$

其中各元素为

$$z_{i_1, i_2, \dots, i_{n-1}, i_{n+1}, \dots, i_N} = \sum_{i_n=1}^{I_n} y_{i_1, i_2, \dots, i_N} a_{i_n} \quad (2.2.9)$$

表 2.2.1 中给出了张量 $\underline{G} \in \mathbb{R}^{J \times R \times P}$ 和矩阵 $\mathbf{A} \in \mathbb{R}^{I \times J}$ 、 $\mathbf{B} \in \mathbb{R}^{T \times R}$ 、 $\mathbf{C} \in \mathbb{R}^{Q \times P}$ 以及向量 $\mathbf{a} \in \mathbb{R}^J$ 、 $\mathbf{b} \in \mathbb{R}^R$ 、 $\mathbf{c} \in \mathbb{R}^P$ 的 n 模乘积规则。

表 2.2.1 n 模乘积规则

n 模乘	矩阵化形式	向量化形式
$\underline{Y} = \underline{G} \times_1 \underline{A} \in \mathbb{R}^{I \times R \times P}$ $y_{irp} = \sum_{j=1}^J g_{jrp} a_{ij}$	$\underline{Y}_{(1)} = \underline{A} \underline{G}_{(1)}$	$\text{vec}(\underline{Y}_{(1)}) = (\underline{I} \otimes \underline{A}) \text{vec}(\underline{G}_{(1)})$
$\underline{Y} = \underline{G} \times_2 \underline{B} \in \mathbb{R}^{J \times T \times P}$ $y_{jtp} = \sum_{r=1}^R g_{jrp} b_{tr}$	$\underline{Y}_{(2)} = \underline{B} \underline{G}_{(2)}$	$\text{vec}(\underline{Y}_{(2)}) = (\underline{I} \otimes \underline{B}) \text{vec}(\underline{G}_{(2)})$
$\underline{Y} = \underline{G} \times_3 \underline{C} \in \mathbb{R}^{J \times R \times Q}$ $y_{jrqp} = \sum_{p=1}^P g_{jrp} c_{qp}$	$\underline{Y}_{(3)} = \underline{C} \underline{G}_{(3)}$	$\text{vec}(\underline{Y}_{(3)}) = (\underline{I} \otimes \underline{C}) \text{vec}(\underline{G}_{(3)})$
$\underline{Y} = \underline{G} \bar{\times}_1 \underline{a} \in \mathbb{R}^{R \times P}$ $y_{rp} = \sum_{j=1}^J g_{jrp} a_j$		$\text{vec}(\underline{Y}_{(1)}) = (\underline{I} \otimes \underline{a}^T) \text{vec}(\underline{G}_{(1)})$ $\text{vec}(\underline{Y}_{(1)}) = \underline{G}_{(1)}^T \underline{a}$
$\underline{Y} = \underline{G} \bar{\times}_2 \underline{b} \in \mathbb{R}^{J \times P}$ $y_{jp} = \sum_{r=1}^R g_{jrp} b_r$	$\underline{Y}_{(2)} = \underline{b}^T \underline{G}_{(2)}$	$\text{vec}(\underline{Y}_{(2)}) = (\underline{I} \otimes \underline{b}^T) \text{vec}(\underline{G}_{(2)})$ $\text{vec}(\underline{Y}_{(2)}) = \underline{G}_{(2)}^T \underline{b}$
$\underline{Y} = \underline{G} \bar{\times}_3 \underline{c} \in \mathbb{R}^{J \times R}$ $y_{jp} = \sum_{p=1}^P g_{jrp} c_p$	$\underline{Y}_{(3)} = \underline{c}^T \underline{G}_{(3)}$	$\text{vec}(\underline{Y}_{(3)}) = (\underline{I} \otimes \underline{c}^T) \text{vec}(\underline{G}_{(3)})$ $\text{vec}(\underline{Y}_{(3)}) = \underline{G}_{(3)}^T \underline{c}$

张量可以和向量进行多模乘积。一个三维张量和向量的两模乘积，其结果为一维张量（向量），一个三阶张量和向量的所有模乘积，其结果为一个标量。我们可以通过以下的规则来交换乘积的阶数：

$$\underline{Y} \bar{\times}_m \underline{a} \bar{\times}_n \underline{b} = (\underline{Y} \bar{\times}_m \underline{a}) \bar{\times}_n \underline{b} = (\underline{Y} \bar{\times}_n \underline{b}) \bar{\times}_m \underline{a}, m < n \quad (2.2.10)$$

例如，张量 $\underline{G} \in \mathbb{R}^{J \times R \times P}$ 和向量 $\underline{a} \in \mathbb{R}^J$ 、 $\underline{b} \in \mathbb{R}^R$ 、 $\underline{c} \in \mathbb{R}^P$ 的 n 模乘积可表示为

$$z = \underline{G} \bar{\times}_1 \underline{a} \bar{\times}_2 \underline{b} \bar{\times}_3 \underline{c} = \sum_{j=1}^J \sum_{r=1}^R \sum_{p=1}^P g_{jrp} a_j b_r c_p$$

更一般地，张量 $\underline{G} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_N}$ 和向量 $\underline{a}^{(n)} \in \mathbb{R}^{J_n}$ 的所有模（ $n=1, 2, \dots, N$ ）的乘积是一个标量：

$$y = \underline{G} \bar{\times}_1 \underline{a}^{(1)} \bar{\times}_2 \underline{a}^{(2)} \dots \bar{\times}_N \underline{a}^{(N)} = \underline{G} \bar{\times} \{\underline{a}\} \in \mathbb{R} \quad (2.2.11)$$

然而，除了模- n 的所有阶的模积结果是一个长度为 J_n 的向量 \underline{x} ：

$$\begin{aligned} \underline{x} &= \underline{G} \bar{\times}_1 \underline{a}^{(1)} \dots \bar{\times}_{n-1} \underline{a}^{(n-1)} \bar{\times}_{n+1} \underline{a}^{(n+1)} \dots \bar{\times}_N \underline{a}^{(N)} \\ &= \underline{G}_{(n)} \left(\underline{a}^{(N)} \otimes \dots \otimes \underline{a}^{(n+1)} \otimes \underline{a}^{(n-1)} \otimes \dots \otimes \underline{a}^{(1)} \right) \\ &= \underline{G} \bar{\times}_n \{\underline{a}\} \in \mathbb{R}^{J_n} \end{aligned} \quad (2.2.12)$$

注意，除了模- n 和模- m 的所有阶模积，其结果是一个 $J_n \times J_m$ 的矩阵。

一个 $I \times J$ 的矩阵 \underline{G} 可以看成是一个第三维为 1 的三阶张量 \underline{G} ，其维数为 $I \times J \times 1$ ，其各模的矩阵化形式为

$$[\underline{\mathbf{G}}]_{(1)} = [\underline{\mathbf{G}}]_{(2)}^T = \mathbf{G} \quad (2.2.13)$$

$$[\underline{\mathbf{G}}]_{(3)} = \text{vec}(\mathbf{G})^T \quad (2.2.14)$$

张量 $\underline{\mathbf{G}}$ 和向量 \mathbf{a} 的模-3 积恰好是 $\underline{\mathbf{G}}$ 和 \mathbf{a} 的外积。

$$\underline{\mathbf{G}} \times_3 \mathbf{a} = \underline{\mathbf{G}} \circ \mathbf{a} \quad (2.2.15)$$

定义 2.2.3: 定义两个相同阶数的张量 $\underline{\mathbf{A}}, \underline{\mathbf{B}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ 的标量积（或内积）为 $\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle$,

所有对应下标元素乘积之和作为标量积的计算结果，即

$$c = \langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle = \sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \cdots \sum_{i_N}^{I_N} b_{i_1, i_2, \dots, i_N} a_{i_1, i_2, \dots, i_N} \in \mathbb{R} \quad (2.2.16)$$

利用标量积可以定义高阶张量 $\underline{\mathbf{A}}$ 的 Frobenius 范数为

$$\|\underline{\mathbf{A}}\|_F = \sqrt{\langle \underline{\mathbf{A}}, \underline{\mathbf{A}} \rangle} = \sqrt{\sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \cdots \sum_{i_N}^{I_N} a_{i_1, i_2, \dots, i_N}^2} \quad (2.2.17)$$

定义张量的 l_1 范数为

$$\|\underline{\mathbf{A}}\|_1 = \sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \cdots \sum_{i_N}^{I_N} |a_{i_1, i_2, \dots, i_N}| \quad (2.2.18)$$

定义 2.2.4: 两个张量 $\underline{\mathbf{A}} \in \mathbb{R}^{I_1 \times \cdots \times I_M \times J_1 \times \cdots \times J_N}$ 和 $\underline{\mathbf{B}} \in \mathbb{R}^{I_1 \times \cdots \times I_M \times K_1 \times \cdots \times K_P}$ 的前 M 阶短缩张量积是一个大小为 $J_1 \times \cdots \times J_N \times K_1 \times \cdots \times K_P$ 的张量：

$$\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle_{1, \dots, M; 1, \dots, M} (j_1, \dots, j_N, k_1, \dots, k_P) = \sum_{i_1=1}^{I_1} \cdots \sum_{i_M=1}^{I_M} a_{i_1, \dots, i_M, j_1, \dots, j_N} b_{i_1, \dots, i_M, k_1, \dots, k_P} \quad (2.2.19)$$

保留的模按照 $\underline{\mathbf{A}}$ 在前 $\underline{\mathbf{B}}$ 在后的次序排列，指定的 $\underline{\mathbf{A}}$ 和 $\underline{\mathbf{B}}$ 的短缩模参数不必是连续的，但是，对应的维数大小必须相等，例如，张量 $\underline{\mathbf{A}} \in \mathbb{R}^{3 \times 4 \times 5}$ 的模-2 和张量 $\underline{\mathbf{B}} \in \mathbb{R}^{7 \times 8 \times 4}$ 的模-3 的短缩张量积，其结果为一个张量 $\underline{\mathbf{C}} = \langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle_{2,3} \in \mathbb{R}^{3 \times 5 \times 7 \times 8}$ 。

$\underline{\mathbf{A}}$ 和 $\underline{\mathbf{B}}$ 的等 M 阶短缩张量积可简化为

$$\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle_{1, \dots, M; 1, \dots, M} = \langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle_{1, \dots, M} \quad (2.2.20)$$

而张量 $\underline{\mathbf{A}} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ 和 $\underline{\mathbf{B}} \in \mathbb{R}^{J_1 \times \cdots \times J_N}$ 除了模- n 的所有模的短缩张量积可表示为

$$\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle_{-n} = \mathbf{A}_{(n)} \mathbf{B}_{(n)}^T \in \mathbb{R}^{I_n \times J_n}, \quad I_k = J_k, \forall k \neq n \quad (2.2.21)$$

张量-向量、张量-矩阵，以及标量的乘积可以表示成缩短积的形式，例如，张量 $\underline{\mathbf{A}}$ 的模- n 和矩阵 $\mathbf{C} \in \mathbb{R}^{J \times I_n}$ 的模-2 短缩积可以通过排列 $\underline{\mathbf{A}}$ 和 \mathbf{C} 的 n 模乘积的维度来得到。

$$\langle \underline{\mathbf{A}}, \mathbf{C} \rangle_{n,2} = \langle \underline{\mathbf{A}}, \mathbf{C}^T \rangle_{n,1} = \text{permute}(\underline{\mathbf{A}} \times_n \mathbf{C}, [1, \dots, n-1, n+1, \dots, N, n]) \quad (2.2.22)$$

同时还有

$$\langle \mathbf{C}, \underline{\mathbf{A}} \rangle_{2,n} = \langle \mathbf{C}^T, \underline{\mathbf{A}} \rangle_{1,n} = \text{permute}(\underline{\mathbf{A}} \times_n \mathbf{C}, [n, 1, \dots, n-1, n+1, \dots, N]) \quad (2.2.23)$$

对于维度相同的张量，它们所有模的短缩积是它们的内积。

$$\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle_{1, \dots, N} = \langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle \quad (2.2.24)$$

在 $M=0$ 的特殊情况下，短缩积变成了两个张量的外积。

2.2.2 张量的特殊形式

张量具有特殊的形式或结构，例如，一个张量是稀疏的或者对称的。

1. 秩-1 张量

张量的秩可以利用外积定义成以下形式（图 2.2.1）^[3]。

定义 2.2.5（秩-1 张量）：如果一个 N 阶张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ 可以写成以下 N 个向量的外积，即

$$\underline{\mathbf{Y}} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \dots \circ \mathbf{a}^{(N)} \quad (2.2.25)$$

那么，该张量是秩-1 张量。

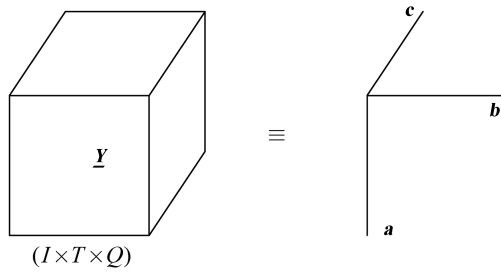


图 2.2.1 秩-1 的三阶张量： $\underline{\mathbf{Y}} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \in \mathbb{R}^{I \times T \times Q}$, $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{b} \in \mathbb{R}^T$, $\mathbf{c} \in \mathbb{R}^Q$

其中 $\mathbf{a}^{(n)} \in \mathbb{R}^{I_n}$ ， $y_{i_1, i_2, \dots, i_N} = a_{i_1}^{(1)} a_{i_2}^{(2)} \dots a_{i_N}^{(N)}$ 。张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ 的秩定义为张量 $\underline{\mathbf{Y}}_1, \dots, \underline{\mathbf{Y}}_R$ 的秩-1 个数的最小值，即 $\underline{\mathbf{Y}} = \sum_{r=1}^R \underline{\mathbf{Y}}_r$ 。

外积通常通过基于以下关系的 Khatri-Rao 积或 Kronecker 积来计算：

$$\text{vec}(\underline{\mathbf{Y}}) = \text{vec}(\underline{\mathbf{Y}}_{(1)}) = \text{vec}\left(\mathbf{a}^{(1)} \left(\mathbf{a}^{(N)} \odot \dots \odot \mathbf{a}^{(2)}\right)^T\right) = \mathbf{a}^{(N)} \odot \dots \odot \mathbf{a}^{(2)} \odot \mathbf{a}^{(1)} \quad (2.2.26)$$

秩-1 张量有许多有意义的性质，并且在多维分析中起着至关重要的作用^[3-5]。一般来讲，更高阶的张量的秩定义为秩-1 张量的最小个数，这些张量的线性组合得到 $\underline{\mathbf{Y}}$ 。这种通过秩-1 张量的线性组合来表示一个张量的表示方法就叫做平行因子分解，它在一定的条件下保持着唯一性。

2. 对称和超对称张量

对于所有 N 个向量 $\mathbf{a}^{(j)}$ 都等于向量 \mathbf{g} 的特殊情况，它们的外积叫做超对称秩-1 张量，超对称张量的每个模都具有相同的维度。

张量也可以在两个或多个模的情况下(部分)对称。例如,如果一个三维张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I \times I \times Q}$, 前面的切片是对称的, 那么它的模-1 和模-2 都是对称的, 即 $\mathbf{Y}_q = \mathbf{Y}_q^T, \forall q = 1, 2, \dots, Q$ 。

3. 对角张量

一个 N 阶立方体张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$, 如果当且仅当 $i_1 = i_2 = \dots = i_N$ 时, 它的元素 $y_{i_1, i_2, \dots, i_N} \neq 0$, 那么它就是一个对角张量, 我们用 $\underline{\mathbf{I}}$ 来表示立方体单位张量, 它的超级对角线上为 1, 其他地方全 0, 这个概念可以如图 2.2.2 所示那样被概括和推广。

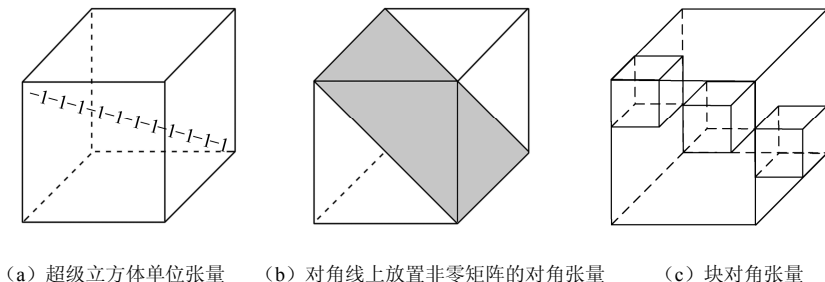


图 2.2.2 三阶张量的特殊形式

2.3 PARAFAC 模型

2.3.1 PARAFAC 模型表示

可以将二维矩阵扩展到三维矩阵情况。令三维矩阵 $\underline{\mathbf{X}} \in \mathbb{C}^{I \times J \times K}$, 元素为 x_{ijk} , 其中, $i = 1, \dots, I, j = 1, \dots, J, k = 1, \dots, K$ 。三维矩阵可以表示成以下向量外积的形式:

$$\underline{\mathbf{X}} = \mathbf{a}_1 \circ \mathbf{b}_1 \circ \mathbf{c}_1 + \dots + \mathbf{a}_R \circ \mathbf{b}_R \circ \mathbf{c}_R = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \quad (2.3.1)$$

其中, $\mathbf{a}_r \in \mathbb{C}^I, \mathbf{b}_r \in \mathbb{C}^J, \mathbf{c}_r \in \mathbb{C}^K, r = 1, \dots, R$ 。式 (2.3.1) 给出了三维矩阵低秩分解过程, 三维矩阵 $\underline{\mathbf{X}}$ 的秩为 R 。Harshman 将式 (2.3.1) 给出的三维矩阵低秩分解模型命名为 PARAFAC^[4], 该模型在本书中, 也称三线性和多线性模型。 $\underline{\mathbf{X}}$ 可以通过一个立方体表示, 平行因子模型如图 2.3.1 所示。

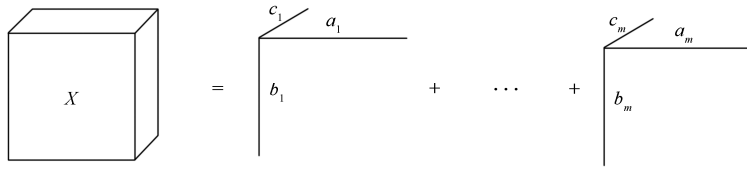


图 2.3.1 平行因子模型

定义三个二维矩阵：

$$\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_R]$$

$$\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_R]$$

$$\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_R]$$

称 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 是平行因子模型的三个承载矩阵，记为

$\underline{x}_{ijk} = [\underline{\mathbf{X}}]_{i,j,k}$ 、 $a_{ir} = [\mathbf{A}]_{i,r}$ 、 $b_{jr} = [\mathbf{B}]_{j,r}$ 、 $c_{kr} = [\mathbf{C}]_{k,r}$ ，PARAFAC 模型的标量形式为

$$\underline{x}_{ijk} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr} \quad (2.3.2)$$

式 (2.3.2) 表明，三维矩阵 $\underline{\mathbf{X}}$ 中的元素 \underline{x}_{ijk} 同样可以表示为 R 个元素乘积之和， \underline{x}_{ijk} 具有三个独立变化的维度，类似地，分别称为“模式 A”、“模式 B”和“模式 C”。PARAFAC 模型是三线性分解模型的一种，它可以看成二维矩阵低秩分解向三维矩阵的扩展。

PARAFAC 模型也可表示为

$$\underline{\mathbf{X}} = \underline{\mathbf{I}} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$$

可以将矩阵低秩分解过程扩展到更高维的情况。记 N 维矩阵 $\underline{\mathbf{X}} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ ，它可以 使用一个超立方体表示，其元素为 $\underline{x}_{i_1 i_2, \dots, i_N}$ ，对 $\underline{\mathbf{X}}$ 进行 N 维低秩分解，有

$$\underline{\mathbf{X}} = \mathbf{a}_1^{(1)} \circ \mathbf{a}_1^{(2)} \circ \dots \circ \mathbf{a}_1^{(N)} + \dots + \mathbf{a}_R^{(1)} \circ \mathbf{a}_R^{(2)} \circ \dots \circ \mathbf{a}_R^{(N)} = \sum_{r=1}^R \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \dots \circ \mathbf{a}_r^{(N)} \quad (2.3.3)$$

其中， $\mathbf{a}_r^{(n)} \in \mathbb{C}^{I_n}$ $r=1, \dots, R, n=1, \dots, N$ ，称式 (2.3.3) 为 N 维平行因子模型，其标量形式为

$$\underline{x}_{i_1 i_2, \dots, i_N} = \sum_{r=1}^R a_{i_1 r}^{(1)} a_{i_2 r}^{(2)} \dots a_{i_N r}^{(N)} \quad (2.3.4)$$

N 维矩阵的元素具有 N 个独立变化的自由度，即具有 N 个“模式”。高维矩阵形式抽象，处理过程较为复杂，本章仅讨论三维矩阵。

2.3.2 PARAFAC 模型的其他表示形式

在实际应用过程中，通常将待处理的数据建模称为 PARAFAC 模型，然后利用低秩分解的思想对数据进行分析和处理。然而，对三维矩阵数据进行处理会带来不便，因此，在进行数据处理的过程中，通常对三维矩阵 $\underline{\mathbf{X}}$ 进行截取剖面，将其表示为等价的二维矩阵形式，下面简要介绍三维矩阵在三个不同的方向的平行剖面截取方法和等价二维矩阵的表示

形式。首先，沿 y 轴对立方体矩阵 $\underline{\mathbf{X}}$ 进行剖面操作，如图 2.3.2 所示。

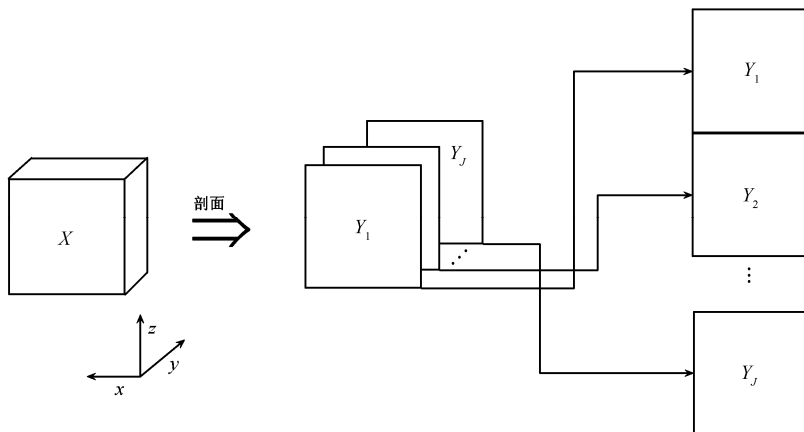


图 2.3.2 PARAFAC 模型的切面形式

其中， $\mathbf{Y}_j^{K \times I}$ 是 $\underline{\mathbf{X}}$ 沿 y 轴的第 j 个子剖面，具有以下形式：

$$\mathbf{Y}_j = \underline{\mathbf{X}}(:, j, :) = \mathbf{C}D_j(\mathbf{B})\mathbf{A}^T \quad (2.3.5)$$

$D_j(\cdot)$ 是抽取矩阵的第 j 行并将其构造为一个对角矩阵。 \mathbf{Y}_j 中的元素与 $\underline{\mathbf{X}}$ 中的元素具有以下对应关系：

$$[\mathbf{Y}_j]_{k,i} = \underline{x}_{ijk} \quad (2.3.6)$$

按照图 2.3.2 的方式，将 J 个子剖面矩阵 $\mathbf{Y}_1, \dots, \mathbf{Y}_J$ 按行排列，得到 y 轴的剖面矩阵 $\mathbf{Y}^{JK \times I}$

$$\mathbf{Y}^{JK \times I} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_J \end{bmatrix} = \begin{bmatrix} \mathbf{C}D_1(\mathbf{B})\mathbf{A}^T \\ \mathbf{C}D_2(\mathbf{B})\mathbf{A}^T \\ \vdots \\ \mathbf{C}D_J(\mathbf{B})\mathbf{A}^T \end{bmatrix} = \begin{bmatrix} \mathbf{C}D_1(\mathbf{B}) \\ \mathbf{C}D_2(\mathbf{B}) \\ \vdots \\ \mathbf{C}D_J(\mathbf{B}) \end{bmatrix} \mathbf{A}^T = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T \quad (2.3.7)$$

矩阵 \mathbf{Y} 中的元素与三维数据 $\underline{\mathbf{X}}$ 中的元素具有以下对应关系：

$$[\mathbf{Y}]_{(j-1)K+k,i} = \underline{x}_{ijk} \quad (2.3.8)$$

同样，按图 2.3.2 所示的截取剖面的方法分别对 $\underline{\mathbf{X}}$ 沿 x 轴和 z 轴截取剖面，得到 I 个 x 轴的子剖面 $\mathbf{X}_i^{J \times K}$ ， $i=1, \dots, I$ 和 K 个 z 轴的子剖面 $\mathbf{Z}_k^{I \times J}$ ， $k=1, \dots, K$ ，它们具有下面的形式：

$$\mathbf{X}_i = \mathbf{B}D_i(\mathbf{A})\mathbf{C}^T, \quad i=1, \dots, I \quad (2.3.9)$$

$$\mathbf{Z}_k = \mathbf{A}D_k(\mathbf{C})\mathbf{B}^T, \quad k=1, \dots, K \quad (2.3.10)$$

其中

$$[\mathbf{X}_i]_{j,k} = \underline{x}_{ijk} \quad (2.3.11)$$

$$[\mathbf{Z}_k]_{i,j} = \underline{x}_{ijk} \quad (2.3.12)$$

将 I 个 x 轴的子剖面 $\mathbf{X}_1, \dots, \mathbf{X}_I$ 按行排列, 可以得到 x 轴的剖面矩阵 $\mathbf{X}^{IJ \times K}$:

$$\mathbf{X}^{IJ \times K} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_I \end{bmatrix} = \begin{bmatrix} \mathbf{B}D_1(\mathbf{A})\mathbf{C}^T \\ \mathbf{B}D_2(\mathbf{A})\mathbf{C}^T \\ \vdots \\ \mathbf{B}D_I(\mathbf{A})\mathbf{C}^T \end{bmatrix} = \begin{bmatrix} \mathbf{B}D_1(\mathbf{A}) \\ \mathbf{B}D_2(\mathbf{A}) \\ \vdots \\ \mathbf{B}D_I(\mathbf{A}) \end{bmatrix} \mathbf{C}^T = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T \quad (2.3.13)$$

\mathbf{X} 中的元素与 $\underline{\mathbf{X}}$ 中的元素具有以下对应关系:

$$[\mathbf{X}]_{(i-1)J+j,k} = \underline{x}_{ijk} \quad (2.3.14)$$

同样, 将 K 个 z 轴的子剖面 $\mathbf{Z}_1, \dots, \mathbf{Z}_K$ 按行排列, 可以得到 z 轴的剖面矩阵 $\mathbf{Z}^{KI \times J}$:

$$\mathbf{Z}^{KI \times J} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \\ \vdots \\ \mathbf{Z}_K \end{bmatrix} = \begin{bmatrix} \mathbf{A}D_1(\mathbf{C})\mathbf{B}^T \\ \mathbf{A}D_2(\mathbf{C})\mathbf{B}^T \\ \vdots \\ \mathbf{A}D_K(\mathbf{C})\mathbf{B}^T \end{bmatrix} = \begin{bmatrix} \mathbf{A}D_1(\mathbf{C}) \\ \mathbf{A}D_2(\mathbf{C}) \\ \vdots \\ \mathbf{A}D_K(\mathbf{C}) \end{bmatrix} \mathbf{B}^T = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T \quad (2.3.15)$$

\mathbf{Z} 中的元素与 $\underline{\mathbf{X}}$ 中的元素具有以下对应关系:

$$[\mathbf{Z}]_{(k-1)I+i,j} = \underline{x}_{ijk} \quad (2.3.16)$$

从式 (2.3.8)、式 (2.3.14) 和式 (2.3.16) 可以看出, 三维 $\underline{\mathbf{X}}$ 的三个剖面矩阵 $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ 仅仅是 $\underline{\mathbf{X}}$ 的三种矩阵排列方式, 它们的矩阵元素是完全相同的, 因此, 对 $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ 进行处理等价于对 $\underline{\mathbf{X}}$ 进行处理。在后续的讨论中, 我们仅在数据的建模步骤中使用三维矩阵, 在数据处理的过程中, 通常使用三个剖面矩阵。PARAFAC 模型表示见表 2.3.1。

表 2.3.1 PARAFAC 模型表示^[3]

运算符	公 式
外积	$\underline{\mathbf{Y}} = \sum_{j=1}^J \mathbf{a}_j \circ \mathbf{b}_j \circ \mathbf{c}_j + \underline{\mathbf{E}}$
标量	$y_{iaq} = \sum_{j=1}^J a_{ij} b_{jq} c_{qj} + e_{iaq}$
模- n 积	$\underline{\mathbf{Y}} = \mathbf{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} + \underline{\mathbf{E}}$
切片表示	$\mathbf{Y}_{:,q} = \mathbf{A}D_q(\mathbf{C})\mathbf{B}^T + \mathbf{E}_{:,q}$ $\mathbf{Y}_{i,:} = \mathbf{B}D_i(\mathbf{A})\mathbf{C}^T + \mathbf{E}_{i,:}$ $\mathbf{Y}_{:,x} = \mathbf{A}D_x(\mathbf{B})\mathbf{C}^T + \mathbf{E}_{:,x}$
矢量	$\text{vec}(\underline{\mathbf{Y}}) = (\mathbf{C} \odot \mathbf{B} \odot \mathbf{A})\mathbf{1} + \text{vec}(\underline{\mathbf{E}}_{(1)})$
Khatri-Rao 积	$\mathbf{Y}_{(1)} \cong \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T + \mathbf{E}_{(1)}$ $\mathbf{Y}_{(2)} \cong \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T + \mathbf{E}_{(2)}$ $\mathbf{Y}_{(3)} \cong \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T + \mathbf{E}_{(3)}$

2.4 PARAFAC 分解唯一性

2.4.1 矩阵本质相等

定义 2.4.1: 矩阵本质相等 (Matrix Essential Equalization) [5], 记矩阵 $\mathbf{A} \in \mathbb{C}^{I \times J}$, 若矩阵 $\bar{\mathbf{A}} = \mathbf{A}\mathbf{\Pi}\mathbf{\Delta}$, 称 \mathbf{A} 与 $\bar{\mathbf{A}}$ 是本质相等的, 记为 $\mathbf{A} \cong \bar{\mathbf{A}}$ 。其中, $\mathbf{\Pi}$ 是列交换矩阵, $\mathbf{\Delta}$ 为对角尺度矩阵。

定义 2.4.1 中的列交换矩阵 $\mathbf{\Pi}$ 的每一行和每一列有一个且只有一个非零元素 1, 例如, 一个 4×4 的列交换矩阵具有下面的形式:

$$\mathbf{\Pi} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

列交换矩阵的作用是将 \mathbf{A} 的列向量按照 $\mathbf{\Pi}$ 的顺序重新排列, 但不改变列向量中元素的值。对角尺度矩阵 $\mathbf{\Delta}$ 是一个 $J \times J$ 的对角矩阵, 具有非零的对角元素。 $\mathbf{\Delta}$ 的作用是将矩阵 \mathbf{A} 的每一列乘上一个非零的幅值。

根据矩阵本质相等的概念, 以二维矩阵为例, 记为

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T \quad (2.4.1)$$

其中 $\mathbf{A} \in \mathbb{C}^{M \times F}$, $\mathbf{B} \in \mathbb{C}^{N \times F}$ 。对于任意的矩阵 $\bar{\mathbf{A}} \in \mathbb{C}^{M \times F}$, $\bar{\mathbf{B}} \in \mathbb{C}^{N \times F}$, 若满足

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T = \bar{\mathbf{A}}\bar{\mathbf{B}}^T$$

则有

$$\bar{\mathbf{A}} = \mathbf{A}\mathbf{\Pi}_1\mathbf{\Delta}_1, \bar{\mathbf{B}} = \mathbf{B}\mathbf{\Pi}_2\mathbf{\Delta}_2 \quad (2.4.2)$$

其中, $\mathbf{\Pi}_1$ 和 $\mathbf{\Pi}_2$ 为列交换矩阵, 表示将 \mathbf{A}, \mathbf{B} 矩阵的列重新排列, $\mathbf{\Delta}_1$ 和 $\mathbf{\Delta}_2$ 为对角尺度矩阵, 表示将 \mathbf{A}, \mathbf{B} 矩阵的每一列分别乘以一个非零系数。此时, 称矩阵分解是唯一的。

由定义 2.4.1 可以看出, 在矩阵分解唯一的情况下, 通过矩阵分解得到的矩阵 $\bar{\mathbf{A}}, \bar{\mathbf{B}}$ 与原矩阵 \mathbf{A}, \mathbf{B} 并不完全相等, 它们仅为本质相等关系, 即

$$\begin{aligned} \bar{\mathbf{A}} &= \mathbf{A}\mathbf{\Pi}_1\mathbf{\Delta}_1 \cong \mathbf{A} \\ \bar{\mathbf{B}} &= \mathbf{B}\mathbf{\Pi}_2\mathbf{\Delta}_2 \cong \mathbf{B} \end{aligned} \quad (2.4.3)$$

根据定义 2.4.1, 由于存在列交换矩阵 $\mathbf{\Pi}_A, \mathbf{\Pi}_B$ 和对角尺度矩阵 $\mathbf{\Delta}_A, \mathbf{\Delta}_B$, 矩阵 $\bar{\mathbf{A}}, \bar{\mathbf{B}}$ 中列向量的排列顺序和幅度可以与 \mathbf{A}, \mathbf{B} 不同。在矩阵理论中, 习惯称为列模糊和尺度模糊, 它们分别由列交换矩阵 $\mathbf{\Pi}$ 和对角尺度矩阵 $\mathbf{\Delta}$ 表征。在矩阵分解的过程中, 若没有对矩阵 \mathbf{A}, \mathbf{B} 进行结构上的约束, 列模糊和尺度模糊是无法避免的。上述问题可以通过矩阵分解的向量形式进行解释, 记 $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_F]$, $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_F]$, 其中, $\mathbf{a}_f \in \mathbb{C}^{I \times 1}$ 、 $\mathbf{b}_f \in \mathbb{C}^{J \times 1}$ 、 $f = 1, \dots, F$ 分别为 \mathbf{A}, \mathbf{B} 的列向量, 则式 (2.4.1) 可以表示为下面的向量形式:

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T = \mathbf{a}_1\mathbf{b}_1^T + \mathbf{a}_2\mathbf{b}_2^T + \cdots + \mathbf{a}_F\mathbf{b}_F^T \quad (2.4.4)$$

其中, $\mathbf{a}_1\mathbf{b}_1^T, \dots, \mathbf{a}_F\mathbf{b}_F^T$ 均为 F 个秩为 1 的矩阵, 此时, 若任意改变 $\mathbf{a}_1\mathbf{b}_1^T, \dots, \mathbf{a}_F\mathbf{b}_F^T$ 的顺序, 矩阵 \mathbf{X} 的值是不变的, 同样, 若向量 \mathbf{a}_f 乘上一个非零系数 λ_f , 同时对应的 \mathbf{b}_f 乘上一个非零系数 $1/\lambda_f$, \mathbf{X} 的值同样不会发生改变。假设将式(2.4.4)中的 $\mathbf{a}_1\mathbf{b}_1^T$ 和 $\mathbf{a}_2\mathbf{b}_2^T$ 交换顺序, 式(2.4.4)可以改写为下列形式:

$$\begin{aligned} \mathbf{X} &= \mathbf{A}\mathbf{B}^T \\ &= \mathbf{a}_1\mathbf{b}_1^T + \mathbf{a}_2\mathbf{b}_2^T + \cdots + \mathbf{a}_F\mathbf{b}_F^T \\ &= \lambda_2\mathbf{a}_2\frac{1}{\lambda_2}\mathbf{b}_2^T + \lambda_1\mathbf{a}_1\frac{1}{\lambda_1}\mathbf{b}_1^T + \cdots + \lambda_F\mathbf{a}_F\frac{1}{\lambda_F}\mathbf{b}_F^T \\ &= [\lambda_2\mathbf{a}_2, \lambda_1\mathbf{a}_1, \dots, \lambda_F\mathbf{a}_F] \begin{bmatrix} \frac{1}{\lambda_2}\mathbf{b}_2^T \\ \frac{1}{\lambda_1}\mathbf{b}_1^T \\ \vdots \\ \frac{1}{\lambda_F}\mathbf{b}_F^T \end{bmatrix} \\ &= \mathbf{A} \begin{bmatrix} 0 & 1 & 0, \dots, 0 \\ 1 & 0 & 0, \dots, 0 \\ \vdots & & \\ 0 & 0 & 0, \dots, 1 \end{bmatrix} \begin{bmatrix} \lambda_2 & & & \\ & \lambda_1 & & \\ & & \ddots & \\ & & & \lambda_F \end{bmatrix} \mathbf{B} \begin{bmatrix} 0 & 1 & 0, \dots, 0 \\ 1 & 0 & 0, \dots, 0 \\ \vdots & & \\ 0 & 0 & 0, \dots, 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\lambda_2} & & & \\ & \frac{1}{\lambda_1} & & \\ & & \ddots & \\ & & & \frac{1}{\lambda_F} \end{bmatrix}^T \\ &= \mathbf{A}\mathbf{\Pi}_A\mathbf{A}_A(\mathbf{B}\mathbf{\Pi}_B\mathbf{A}_B)^T \\ &= \bar{\mathbf{A}}\bar{\mathbf{B}}^T \end{aligned} \quad (2.4.5)$$

得到

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T = \bar{\mathbf{A}}\bar{\mathbf{B}}^T$$

其中

$$\begin{aligned} \bar{\mathbf{A}} &= \mathbf{A}\mathbf{\Pi}_A\mathbf{A}_A \\ \bar{\mathbf{B}} &= \mathbf{B}\mathbf{\Pi}_B\mathbf{A}_B \end{aligned}$$

可以看出, 总存在 $\bar{\mathbf{A}}$ 和 $\bar{\mathbf{B}}$ 实现矩阵分解, 但它们仅与 \mathbf{A}, \mathbf{B} 是本质相等的。因此, 列模糊和尺度模糊是矩阵分解过程中固有的模糊, 在没有额外约束条件的情况下, 通过矩阵分解, 无法确定承载矩阵列的顺序和幅度。因此, **定义 2.4.1** 给出的矩阵分解唯一性又可以称为“本质唯一性”。在实际应用过程中矩阵分解所带来的列模糊和尺度模糊可以通过一些方法消除。

2.4.2 二维矩阵低秩分解不唯一性

当二维矩阵的秩大于 1 时, 若没有特殊的结构约束, 二维矩阵低秩分解是不唯一的, 即对于矩阵分解过程

$$X = AB^T$$

存在另一组矩阵 \bar{A}, \bar{B} , 使得

$$X = \bar{A}\bar{B}^T$$

但

$$\bar{A} \neq A\Pi_A A_A, \quad \bar{B} \neq B\Pi_B A_B$$

其中, $\Pi_{A,B}, A_{A,B}$ 是定义 2.4.1 中给出的列交换矩阵和对角尺度矩阵。二维矩阵分解的不唯一性可以通过反证法说明, 给出任意的满秩方阵 $T \in \mathbb{C}^{F \times F}$, 有

$$X = AB^T = ATT^{-1}B^T = \bar{A}\bar{B}^T$$

其中

$$\bar{A} = AT = [\bar{a}_1, \dots, \bar{a}_F]$$

$$\bar{B} = B(T^{-1})^T = [\bar{b}_1, \dots, \bar{b}_F]$$

式中 \bar{a}_f, \bar{b}_f 分别是 A, B 的列向量。若 A, B 为满秩矩阵, 则 \bar{A}, \bar{B} 亦为满秩矩阵, 则

$$X = \bar{A}\bar{B}^T = \bar{a}_1\bar{b}_1^T + \bar{a}_2\bar{b}_2^T + \dots + \bar{a}_F\bar{b}_F^T$$

满足低秩分解的定义, 但 $T \neq \Pi A$ 。因此, 二维矩阵低秩分解不唯一。然而, 在二维矩阵分解的过程中, 若承载矩阵 A, B 满足一定的结构约束, 矩阵可以实现分解唯一, 如奇异值分解等^[5]。

2.4.3 PARAFAC 分解唯一性

PARAFAC 模型分解与二维矩阵低秩分解的根本不同在于, PARAFAC 分解具有唯一性, 这是 PARAFAC 模型广泛应用于数据分析的重要原因之一。在允许列模糊和尺度模糊的前提下, 仅知三维矩阵 X , 可以唯一地得到承载矩阵 A, B, C 。PARAFAC 模型分解的唯一性定理最早由 J.B.Kruskal 提出^[7], 对于实数情况下的 PARAFAC 模型的唯一性进行了证明。N. D. Sidiropoulos 对其进行了发展, 给出了复数域内 PARAFAC 模型的唯一性定理^[8], 为 PARAFAC 模型在信号处理中的应用奠定了基础。PARAFAC 模型唯一性定理中需要使用矩阵 k -秩的概念, 这同样是 Kruskal 在文献^[6]中定义的。

定理 2.4.1: PARAFAC 唯一性定理^[6], 考虑 PARAFAC 模型沿 X 轴的 I 个子剖面矩阵:

$$X_i^{J \times K} = B D_i(A) C^T, i=1, \dots, I$$

其中, $A \in \mathbb{R}^{I \times R}$, $B \in \mathbb{R}^{J \times R}$, $C \in \mathbb{R}^{K \times R}$, 如果满足

$$k_A + k_B + k_C \geq 2(R+1)$$

则 A, B, C 在存在尺度模糊和列模糊的条件下是唯一的。即, 若存在矩阵组 \hat{A} , \hat{B} 和 \hat{C} , 使得

$$\mathbf{X}_i^{J \times K} = \hat{\mathbf{B}} \mathbf{D}_i(\hat{\mathbf{A}}) \hat{\mathbf{C}}^T, i=1, \dots, I$$

则 $\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}$ 和 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 之间有以下关系:

$$\hat{\mathbf{A}} = \mathbf{A} \mathbf{\Pi} \mathbf{\Lambda}_1, \hat{\mathbf{B}} = \mathbf{B} \mathbf{\Pi} \mathbf{\Lambda}_2, \hat{\mathbf{C}} = \mathbf{C} \mathbf{\Pi} \mathbf{\Lambda}_3 \quad (2.4.6)$$

$\mathbf{\Pi}$ 是列模糊矩阵, $\mathbf{\Lambda}_1, \mathbf{\Lambda}_2$ 和 $\mathbf{\Lambda}_3$ 是尺度模糊矩阵, 且 $\mathbf{\Lambda}_1 \mathbf{\Lambda}_2 \mathbf{\Lambda}_3 = \mathbf{I}$ 。

PARAFAC 的唯一性定理是本书研究的基础, 贯穿本书始终。在不同的应用环境中, 唯一性定理会发生改变, 在后续的讨论中, 我们会给出具体情况下的 PARAFAC 模型唯一性定理。

2.5 本章小结

本章给出了本书常用的矩阵代数和张量代数知识, 通过矩阵低秩分解引出了平行因子模型, 给出了平行因子模型的各种形式。分析了二维矩阵分解的不唯一性, 介绍了平行因子模型的唯一性理论, 给出了唯一性定理。本章我们参考了很多文献, 如文献[2,3,5], 对相关作者表示感谢。本章的概念和定理是本书后续部分的基础, 在此简述以做铺垫。

参考文献

- [1] 张小飞, 汪飞, 徐大专. 阵列信号处理的理论和应用[M]. 北京: 国防工业出版社, 2010.
- [2] 张小飞, 汪飞, 陈华伟, 徐大专. 阵列信号处理的理论和应用第二版[M]. 北京: 国防工业出版社, 2013.
- [3] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, et al, Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation[J]. Wiley 2009.
- [4] R. A. Harshman, Foundations of the PARAFAC Procedure: Models and Conditions for an “Explanatory” Multimodal Factor Analysis[J]. UCLC Working Papers in Phonetics, 1970, 16: 1-84.
- [5] 刘旭. 基于多维矩阵低秩分解的信号处理技术研究[D]. 博士学位论文, 南京航空航天大学, 2009.
- [6] 张贤达, 保铮. 通信信号处理[M]. 北京: 国防工业出版社, 2000.
- [7] J. B. Kruskal, Three-way arrays: Rank and Uniqueness of Trilinear Decompositions, with Application to Arithmetic Complexity and Statistics[J]. Linear Algebra Applications, 1977 18: 95-138.
- [8] N. D. Sidiropoulos, G. B. Giannakis and R. Bro, Blind PARAFAC Receivers for DS-CDMA Systems[J]. IEEE Transactions on Signal Processing, 2000, 48 (3): 810-823.

第3章

PARAFAC 基本理论

本章研究 PARAFAC 模型、 k -秩、可辨识性、PARAFAC 分解、PARAFAC 分解的 CRB 分析、自适应 PARAFAC 分解和大规模 PARAFAC 分解。

3.1 PARAFAC 模型

3.1.1 三线性模型

三线性分解又称规范分解、平行因子 (PARAFAC) 分析。 $I \times J \times K$ 的三维阵列 $\underline{\mathbf{X}}$ (其元素为 $x_{i,j,k}$) 以及 F 元的三线性分解^[1-6]:

$$x_{i,j,k} = \sum_{f=1}^F a_{i,f} b_{j,f} c_{k,f} \quad (3.1.1)$$

其中 $i=1,2,\dots,I, j=1,2,\dots,J, k=1,2,\dots,K$ 。定义下列矩阵: $I \times F$ 维矩阵 \mathbf{A} , 其元素为 $\mathbf{A}(i,f)=a_{i,f}$; $J \times F$ 维矩阵 \mathbf{B} , 其元素为 $\mathbf{B}(j,f)=b_{j,f}$; $K \times F$ 维矩阵 \mathbf{C} , 其元素为 $\mathbf{C}(k,f)=c_{k,f}$ 。此外, 定义了 $J \times K$ 维矩阵 \mathbf{X}_i , $K \times I$ 维矩阵 \mathbf{X}_j 和 $I \times J$ 维矩阵 \mathbf{X}_k , 其相应的典型元素为 $\mathbf{X}_i(j,k)=\mathbf{X}_j(k,i)=\mathbf{X}_k(i,j)=x_{i,j,k}$, 则式 (3.1.1) 中的模型可以表示为沿三个不同维度去“切”三维阵列 \mathbf{X} 的结果:

$$\mathbf{X}_i = \mathbf{B} \mathbf{D}_i(\mathbf{A}) \mathbf{C}^T, i=1,2,\dots,I \quad (3.1.2)$$

$$\mathbf{X}_j = \mathbf{C} \mathbf{D}_j(\mathbf{B}) \mathbf{A}^T, j=1,2,\dots,J \quad (3.1.3)$$

$$\mathbf{X}_k = \mathbf{A} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T, k=1, 2, \dots, K \quad (3.1.4)$$

其中 $\mathbf{D}_i(\mathbf{A})$ 为由矩阵 \mathbf{A} 的第 i 行元素构成的对角矩阵。PARAFAC 是一个三维模型，属于多维阵列的代数，也称多维分析。正如奇异值分解可以看成矩阵的低秩分解一样，PARAFAC 可以看成三维阵列的低秩分解。矩阵的低秩矩阵分解不是唯一的（奇异值分解之所以是唯一的，是因为应用了正交性约束），但在适当的假设条件下，PARAFAC 分解是唯一的，不需要正交性或其他约束条件。

通过矩阵重构，三维阵列 \mathbf{X} 可以表示为二维矩阵的形式。根据式 (3.1.2)，构造以下矩阵：

$$\mathbf{X}^{(KJ \times I)} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_J \end{bmatrix} = \begin{bmatrix} \mathbf{C} \mathbf{D}_1(\mathbf{B}) \mathbf{A}^T \\ \mathbf{C} \mathbf{D}_2(\mathbf{B}) \mathbf{A}^T \\ \vdots \\ \mathbf{C} \mathbf{D}_J(\mathbf{B}) \mathbf{A}^T \end{bmatrix} = [\mathbf{B} \odot \mathbf{C}] \mathbf{A}^T \quad (3.1.5)$$

根据式 (3.1.3)，构造以下矩阵：

$$\mathbf{X}^{(IJ \times K)} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_I \end{bmatrix} = \begin{bmatrix} \mathbf{B} \mathbf{D}_1(\mathbf{A}) \mathbf{C}^T \\ \mathbf{B} \mathbf{D}_2(\mathbf{A}) \mathbf{C}^T \\ \vdots \\ \mathbf{B} \mathbf{D}_I(\mathbf{A}) \mathbf{C}^T \end{bmatrix} = [\mathbf{A} \odot \mathbf{B}] \mathbf{C}^T \quad (3.1.6)$$

根据式 (3.1.4)，构造以下矩阵：

$$\mathbf{X}^{(KL \times J)} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_K \end{bmatrix} = \begin{bmatrix} \mathbf{A} \mathbf{D}_1(\mathbf{C}) \mathbf{B}^T \\ \mathbf{A} \mathbf{D}_2(\mathbf{C}) \mathbf{B}^T \\ \vdots \\ \mathbf{A} \mathbf{D}_K(\mathbf{C}) \mathbf{B}^T \end{bmatrix} = [\mathbf{C} \odot \mathbf{A}] \mathbf{B}^T \quad (3.1.7)$$

3.1.2 四线性模型或多线性模型

考虑一个 $I \times J \times K \times L$ 的四线性模型 \mathbf{X} ，其元素表示为^[1-6]

$$x_{i,j,k,l} = \sum_{f=1}^F a_{i,f} b_{j,f} g_{k,f} h_{l,f} \quad (3.1.8)$$

其中 $i=1, \dots, I, j=1, \dots, J, k=1, \dots, K, l=1, \dots, L$ 。定义一个 $K \times F$ 的矩阵 \mathbf{G} ，其元素 $\mathbf{G}(k, f) := g_{k,f}$ ，定义一个 $L \times F$ 的矩阵 \mathbf{H} ，其元素 $\mathbf{H}(l, f) := h_{l,f}$ 。和三线性模型的情况类似，四线性模型 \mathbf{X} 可表示成各种切片形式：

$$\mathbf{X}_{i,g} = \mathbf{B} \mathbf{D}_i(\mathbf{A}) \mathbf{D}_g(\mathbf{G}) \mathbf{H}^T, i=1, \dots, I, g=1, \dots, G \quad (3.1.9)$$

$$\mathbf{X}_{j,h} = \mathbf{G} \mathbf{D}_j(\mathbf{B}) \mathbf{D}_h(\mathbf{H}) \mathbf{A}^T, j=1, 2, \dots, J, h=1, \dots, H \quad (3.1.10)$$

$$\mathbf{X}_{h,j} = \mathbf{A} \mathbf{D}_h(\mathbf{H}) \mathbf{D}_j(\mathbf{B}) \mathbf{G}^T, h=1, 2, \dots, H, j=1, \dots, J \quad (3.1.11)$$

$$\mathbf{X}_{g,i} = \mathbf{H} \mathbf{D}_g(\mathbf{G}) \mathbf{D}_i(\mathbf{A}) \mathbf{B}^T, \quad g=1, \dots, G, \quad i=1, \dots, I \quad (3.1.12)$$

同样，使用 Khatri-Rao 积可以将式 (3.1.8) 中的模型构造成矩阵形式：

$$\mathbf{X}^{(JLI \times K)} = (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \mathbf{G}^T \quad (3.1.13)$$

$$\mathbf{X}^{(KIJ \times L)} = (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{H}^T \quad (3.1.14)$$

$$\mathbf{X}^{(LJK \times I)} = (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \mathbf{A}^T \quad (3.1.15)$$

$$\mathbf{X}^{(IKL \times J)} = (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \mathbf{B}^T \quad (3.1.16)$$

进一步推广到 N 维，考虑一个 $I_1 \times I_2 \times \dots \times I_N$ 的 N 维数据集 $\underline{\mathbf{X}}$ ，其元素具有以下形式

$$x_{i_1, \dots, i_N} = \sum_{f=1}^F \prod_{n=1}^N a_{i_n, f}^{(n)} \quad (3.1.17)$$

其中 $i_n = 1, \dots, I_n$, $n = 1, \dots, N$, $a_{i_n, f}^{(n)} \in \mathbb{C}$ 。定义 $I_n \times F$ 的矩阵 $\mathbf{A}_{(n)}$, $n = 1, \dots, N$, $\mathbf{A}_{(n)}(i_n, f) := a_{i_n, f}^{(n)}$, N 维数据集 $\underline{\mathbf{X}}$ 可以表示为矩阵形式：

$$\underline{\mathbf{X}}^{(I_2 \dots I_N \times I_1)} = (\mathbf{A}_{(N)} \odot \dots \odot \mathbf{A}_{(2)}) (\mathbf{A}_{(1)})^T \quad (3.1.18)$$

3.2 k -秩

与矩阵秩的概念一样， k -秩 (Kruskal-秩) 的概念在多线性代数里起着非常重要的作用。

定义 3.2.1^[6,7]: 对于给定的矩阵 $\mathbf{A} \in \mathbb{C}^{I \times F}$ ，当且仅当 \mathbf{A} 包含至少 r 个独立的列时， \mathbf{A} 的秩为 $r_A = \text{Rank}(\mathbf{A}) = r$ 。如果矩阵 \mathbf{A} 的任意 k 列独立，则 \mathbf{A} 的 k -秩 $k_A = k$ 。
 $k_A \leq r_A \leq \min(I, F), \forall \mathbf{A}$ 。

性质 3.2.1^[6,7]: 一个随机矩阵 $\mathbf{A} \in \mathbb{C}^{I \times F}$ ，其列是从绝对连续分布中独立提出的，则它具有满秩，并具有满 k -秩，即

$$k_A = r_A = \min(I, F)$$

性质 3.2.2^[6,7]: Vandermonde 矩阵的 k -秩。一个由非零序列 $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{C}$ 构成的 Vandermonde 矩阵：

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \alpha_n^2 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

不仅满秩，而且满 k -秩。

性质 3.2.3^[6,7]: Khatri-Rao 积的 k -秩。考虑 Khatri-Rao (列 Kronecker) 积：

$$\mathbf{B} \odot \mathbf{A} = \begin{bmatrix} \mathbf{AD}_1(\mathbf{B}) \\ \mathbf{AD}_2(\mathbf{B}) \\ \vdots \\ \mathbf{AD}_J(\mathbf{B}) \end{bmatrix}$$

其中, \mathbf{A} 的大小为 $I \times F$, \mathbf{B} 的大小为 $J \times F$ 。如果 \mathbf{A} 和 \mathbf{B} 均不含有全零列 (因此 $k_A \geq 1, k_B \geq 1$), 则 $k_{\mathbf{B} \odot \mathbf{A}} \geq \min(k_A + k_B - 1, F)$ 。

3.3 可辨识性

三线性模型的本质特征是其唯一性。在一定条件下, 三线性模型本质上是唯一的, 即在没有阵列模糊的情况下, \mathbf{A}, \mathbf{B} 和 \mathbf{C} 是可辨识的。下面介绍几个结论^[6-8]。

定理 3.3.1: 给定 $\mathbf{X}_i = \mathbf{BD}_i(\mathbf{A})\mathbf{C}^T$, $i=1, \dots, I$, $\mathbf{A} \in \mathbb{C}^{I \times F}$, $\mathbf{B} \in \mathbb{C}^{J \times F}$, $\mathbf{C} \in \mathbb{C}^{K \times F}$, 如果

$$k_A + k_B + k_C \geq 2F + 2$$

则 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 对于列交换和 (复数) 尺度变换是唯一的。

从绝对连续分布中取出的相对独立的列组成的矩阵具有满 k -秩。如果三个矩阵都满足该条件, 则可辨识的充分条件为

$$\min(I, F) + \min(J, F) + \min(K, F) \geq 2F + 2$$

如果 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 有其他的结构约束, 则可能获得更佳的可辨识性结果。例如, 将 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 中的一个或几个限制为 Vandermonde 矩阵, 存在如下结论。

定理 3.3.2: $\mathbf{X}_i = \mathbf{BD}_i(\mathbf{A})\mathbf{C}^T$, $i=1, \dots, I$, $\mathbf{A} \in \mathbb{C}^{I \times F}$, $\mathbf{B} \in \mathbb{C}^{J \times F}$, $\mathbf{C} \in \mathbb{C}^{K \times F}$, \mathbf{A} 是由非零序列构成的 Vandermonde 矩阵。如果

$$k_B + \min(I + k_C, 2F) \geq 2F + 2$$

则 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 是可辨识的 (列交换和尺度变换)。

如果三个矩阵中有两个以上为 Vandermonde 矩阵, 唯一性结论会进一步增强。

定理 3.3.3: 给定 $\mathbf{X}_i = \mathbf{BD}_i(\mathbf{A})\mathbf{C}^T$, $i=1, \dots, I$, $\mathbf{A} \in \mathbb{C}^{I \times F}$, $\mathbf{B} \in \mathbb{C}^{J \times F}$, $\mathbf{C} \in \mathbb{C}^{K \times F}$, 设 \mathbf{A} 和 \mathbf{B} 是由非零序列构成的 Vandermonde 矩阵, 如果

$$I + J + k_C \geq 2F + 2$$

则 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 是可辨识的 (列交换和尺度变换)。

如果三个矩阵全是 Vandermonde 矩阵, 则可得到下述结论。

定理 3.3.4: 给定的 $\mathbf{X}_i = \mathbf{BD}_i(\mathbf{A})\mathbf{C}^T$, $i=1, \dots, I$, $\mathbf{A} \in \mathbb{C}^{I \times F}$, $\mathbf{B} \in \mathbb{C}^{J \times F}$, $\mathbf{C} \in \mathbb{C}^{K \times F}$, 设 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 是由非零序列构成的 Vandermonde 矩阵。如果

$$I + J + K \geq 2F + 2$$

则 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 是可辨识 (列交换和尺度变换)。

定理 3.3.5: 多线性分解的唯一性。考虑一个 d -线性模型^[6-8]:

$$x_{i_1, \dots, i_d} = \sum_{f=1}^F \prod_{\delta=1}^d a_{i_\delta f}^{(\delta)}$$

其中 $i_\delta = 1, \dots, I_\delta$, $\delta = 1, \dots, d$, $a_{i_\delta f}^{(\delta)} \in \mathbb{C}$, 给定 x_{i_1, \dots, i_d} , $i_\delta = 1, \dots, I_\delta$, $\delta = 1, \dots, d$, 若满足

$$\sum_{\delta=1}^d k_{A^{(\delta)}} \geq 2F + (d-1)$$

则 $A^{(\delta)}$ ($\delta = 1, \dots, d$) 对于列交换和尺度变换是唯一的。

3.4 PARAFAC 分解

3.4.1 三线性交替最小二乘

很多算法可以实现 PARAFAC 分解, 本节主要介绍较为常用的三线性交替最小二乘 (Trilinear Alternating Least Square, TALS) 算法。三线性交替最小二乘算法是三线性模型的数据检测的一种常用方法。TALS 的基本思想很简单, 矩阵更新的方法是: 每一步更新一个矩阵, 对余下的矩阵, 依据前一次估计的结果, 利用最小二乘 (Least Square, LS) 来更新; 继续对其他矩阵进行更新; 重复以上步骤直到算法收敛。TALS 算法是全局单调收敛的^[4-6]。

假设 $I \times J \times K$ 的三维数据集 $\underline{\mathbf{X}}$, 其三线性模型有以下形式:

$$x_{i,j,k} = \sum_{f=1}^F a_{i,f} b_{j,f} c_{k,f} + e_{ijk}, \quad i=1, \dots, I, \quad j=1, \dots, J, \quad k=1, \dots, K \quad (3.4.1)$$

其中, F 表示因子数, $a_{i,f}$ 是矢量 \mathbf{a}_f 中第 i 个元素, $b_{j,f}$ 是矢量 \mathbf{b}_f 中第 j 个元素, $c_{k,f}$ 是矢量 \mathbf{c}_f 中第 k 个元素。 $x_{i,j,k}$ ($i=1, \dots, I; j=1, \dots, J; k=1, \dots, K$) 形成了 $I \times J \times K$ 的三维数据集 $\underline{\mathbf{X}}$ 。 e_{ijk} 为误差, e_{ijk} ($i=1, \dots, I; j=1, \dots, J; k=1, \dots, K$) 形成了 $I \times J \times K$ 的三维误差集 $\underline{\mathbf{E}}$ 。

定义: $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_F]$ 为 $I \times F$ 矩阵, $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_F]$ 为 $J \times F$ 矩阵, $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_F]$ 为 $K \times F$ 矩阵。三线性模型写成切片形式:

$$\tilde{\mathbf{Z}}_j = \mathbf{A} \mathbf{D}_j(\mathbf{B}) \mathbf{C}^T + \mathbf{E}_j, \quad j=1, 2, \dots, J$$

$$\tilde{\mathbf{X}}_k = \mathbf{B} \mathbf{D}_k(\mathbf{C}) \mathbf{A}^T + \mathbf{E}_k, \quad k=1, 2, \dots, K$$

$$\tilde{\mathbf{Y}}_i = \mathbf{C} \mathbf{D}_i(\mathbf{A}) \mathbf{B}^T + \mathbf{E}_i, \quad i=1, 2, \dots, I$$

① 矩阵 \mathbf{A} 的计算:

$$\begin{bmatrix} \tilde{\mathbf{X}}_1 \\ \tilde{\mathbf{X}}_2 \\ \vdots \\ \tilde{\mathbf{X}}_K \end{bmatrix} = \begin{bmatrix} \mathbf{B} \mathbf{D}_1(\mathbf{C}) \\ \mathbf{B} \mathbf{D}_2(\mathbf{C}) \\ \vdots \\ \mathbf{B} \mathbf{D}_K(\mathbf{C}) \end{bmatrix} \mathbf{A}^T + \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \vdots \\ \mathbf{E}_K \end{bmatrix}$$

A^T 的最小二乘的估计:

$$\hat{A}^T = \begin{bmatrix} BD_1(C) \\ BD_2(C) \\ \vdots \\ BD_K(C) \end{bmatrix}^+ \begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \\ \vdots \\ \tilde{X}_K \end{bmatrix}$$

② 矩阵 B 的计算:

$$\begin{bmatrix} \tilde{Y}_1 \\ \tilde{Y}_2 \\ \vdots \\ \tilde{Y}_I \end{bmatrix} = \begin{bmatrix} CD_1(A) \\ CD_2(A) \\ \vdots \\ CD_I(A) \end{bmatrix} B^T + \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_I \end{bmatrix}$$

B^T 的最小二乘的估计:

$$\hat{B}^T = \begin{bmatrix} CD_1(A) \\ CD_2(A) \\ \vdots \\ CD_I(A) \end{bmatrix}^+ \begin{bmatrix} \tilde{Y}_1 \\ \tilde{Y}_2 \\ \vdots \\ \tilde{Y}_I \end{bmatrix}$$

③ 矩阵 C 的计算:

$$\begin{bmatrix} \tilde{Z}_1 \\ \tilde{Z}_2 \\ \vdots \\ \tilde{Z}_J \end{bmatrix} = \begin{bmatrix} AD_1(B) \\ AD_2(B) \\ \vdots \\ AD_J(B) \end{bmatrix} C^T + \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_J \end{bmatrix}$$

C^T 的最小二乘的估计

$$\hat{C}^T = \begin{bmatrix} AD_1(B) \\ AD_2(B) \\ \vdots \\ AD_J(B) \end{bmatrix}^+ \begin{bmatrix} \tilde{Z}_1 \\ \tilde{Z}_2 \\ \vdots \\ \tilde{Z}_J \end{bmatrix}$$

④ 循环①~③, 更新矩阵, 直到收敛。

3.4.2 平行因子的快速算法

1. COMFAC 算法

TALS 算法的复杂度较高, 收敛速度较慢。下面介绍一种提高 TALS 收敛速度的方法, 称为复平行因子分析 (Complex Parallel Factor Analysis, COMFAC) 算法, 其基本步骤为^[9]:

- ① 数据阵压缩;
- ② 平行因子模型对数据阵的初始化和匹配;
- ③ 在原来的空间中, 进行数据阵解压缩。

数据阵压缩。考虑一个三维数据阵 $\mathbf{X}(I \times J \times K)$, 定义压缩矩阵 \mathbf{U} 是 $I \times P$ 维、矩阵 $\mathbf{V}(J \times Q)$ 和 $\mathbf{Z}(K \times R)$, 其中, $P \leq I, Q \leq J, R \leq K$ 。压缩后数据阵 \mathbf{G} 的大小为 $P \times Q \times R$ 。压缩过程如下:

$$\mathbf{G}^{(P \times QR)} = \mathbf{U}^H \mathbf{X}^{(I \times JK)} (\mathbf{Z} \otimes \mathbf{V}) \quad (3.4.2)$$

恢复 $\mathbf{X}^{(I \times JK)}$ 如下:

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{U} \mathbf{G}^{(P \times QR)} (\mathbf{Z}^H \otimes \mathbf{V}^H) \quad (3.4.3)$$

其中压缩矩阵采用 Tucker3-ALS 算法。

步骤 1: 初始化 \mathbf{V} 和 \mathbf{Z} 。

步骤 2: \mathbf{U} 等于 $\mathbf{X}^{(I \times JK)} (\mathbf{Z} \otimes \mathbf{V})$ 的前 P 个左奇异值矢量。

步骤 3: \mathbf{V} 等于 $\mathbf{X}^{(J \times IK)} (\mathbf{Z} \otimes \mathbf{U})$ 的前 Q 个左奇异值矢量。

步骤 4: \mathbf{Z} 等于 $\mathbf{X}^{(K \times IJ)} (\mathbf{V} \otimes \mathbf{U})$ 的前 R 个左奇异值矢量。

步骤 5: 转到步骤 2, 直到收敛。

步骤 6: $\mathbf{G}^{(P \times QR)} = \mathbf{U}^H \mathbf{X}^{(I \times JK)} (\mathbf{Z} \otimes \mathbf{V})$ 。

注意, 更新步骤 2~4 表明, 只有算法收敛, 才需要计算 \mathbf{G} 。Tucker3 模型直接给出了压缩数据阵 \mathbf{G} 。

平行因子模型初始化和匹配: 对 \mathbf{G} 的平行因子模型进行初始化。对每个模式使用不同的奇异值矢量, 对 $\mathbf{G}^{(P \times QR)}$ 、 $\mathbf{G}^{(Q \times RP)}$ 和 $\mathbf{G}^{(R \times PQ)}$ 进行奇异值分解, 取前 F 个左奇异值矢量。

\mathbf{G} 切片形式:

$$\mathbf{G}_r = \mathbf{L} \mathbf{D}_r(\mathbf{N}) \mathbf{M}^T + \mathbf{E}_r, \quad r = 1, \dots, R$$

其中 $\mathbf{L}(P \times F)$ 、 $\mathbf{M}(Q \times F)$ 和 $\mathbf{N}(R \times F)$ 为因子矩阵, \mathbf{E}_r 为误差。这一模型也可以通过 Khatri-Rao 积来表示。采用这些符号后, PARAFAC 模型可表示为

$$\hat{\mathbf{G}}^{(P \times QR)} = \mathbf{L}(\mathbf{N} \odot \mathbf{M})^T \quad (3.4.4)$$

或等价表示为

$$\hat{\mathbf{G}}^{(P \times QR)} = \mathbf{L}(\mathbf{N}^* \odot \mathbf{M}^*)^H \quad (3.4.5)$$

解压缩与细化: 经过压缩空间里的模型匹配之后, 还需要进行解压缩, 其过程如下。可将 \mathbf{G} 变换为原来的域:

$$\hat{\mathbf{X}}^{(I \times JK)} = \mathbf{U} \mathbf{G}^{(P \times QR)} (\mathbf{Z}^H \otimes \mathbf{V}^H) \quad (3.4.6)$$

代入式 (3.4.5) 可得

$$\hat{\mathbf{X}}^{(I \times JK)} \approx \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \quad (3.4.7)$$

由上述推导可得, $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 可近似表示为

$$\mathbf{A} \approx \mathbf{U} \mathbf{L}, \quad \mathbf{B} \approx \mathbf{V}^* \mathbf{M}, \quad \mathbf{C} \approx \mathbf{Z}^* \mathbf{N} \quad (3.4.8)$$

在原来空间分解, 可以用交替最小二乘算法来细化。仅需要较少的迭代次数得到的解

通常非常接近最小二乘解。

2. TALS-LS 算法

由上所述，平行因子的模型可以写成以下矩阵形式：

$$\mathbf{X}^{(I \times JK)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \quad (3.4.9)$$

其中 $\mathbf{A} \in \mathbb{C}^{I \times F}$ ， $\mathbf{B} \in \mathbb{C}^{J \times F}$ ， $\mathbf{C} \in \mathbb{C}^{K \times F}$ ； $\mathbf{X}^{(I \times JK)}$ 是 $\mathbf{X}(I \times J \times K)$ 在第一模式下的变换形式。

TALS 算法就是使下式最小化：

$$\gamma = \left\| \mathbf{X}^{(I \times JK)} - \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \right\|_F^2 \quad (3.4.10)$$

因此，TALS 算法矩阵 \mathbf{A} 的估计可以表示为：

$$\hat{\mathbf{A}} = \mathbf{X}^{(I \times JK)} (\mathbf{Z}_a^+)^T \quad (3.4.11)$$

其中 $\mathbf{Z}_a = \mathbf{C} \odot \mathbf{B}$ 。同理可以用 $\mathbf{Z}_b = \mathbf{A} \odot \mathbf{C}$ 和 $\mathbf{Z}_c = \mathbf{B} \odot \mathbf{A}$ 估计矩阵 \mathbf{B} 和 \mathbf{C} ，重复以上步骤直至收敛。令 it 表示迭代次数， $\gamma^{(it)}$ 表示第 it 次的迭代结果，当 $|\gamma^{(it)} - \gamma^{(it-1)}| \leq \varepsilon$ ，表示算法收敛，其中 ε 为预先定义的收敛精度。选择一个好的初始值可以使收敛速度加快。但是当数据阵有很大的维数，或者其中一维和另一维非常相似的时候，初始化技术不能解决收敛慢的问题。当两个因子线性相关时，会出现收敛速度慢的现象。在 TALS 算法中加入线性搜索可以加快平行因子的收敛速度^[10]，即本节讨论的 TALS-LS 算法(TALS with Line Search)。

在迭代前通过计算一种线性衰退来预测载入因子的值：

$$\mathbf{A}^{(new)} = \mathbf{A}^{(it-2)} + R_{LS}(\mathbf{A}^{(it-1)} - \mathbf{A}^{(it-2)}) \quad (3.4.12)$$

其中 $\mathbf{A}^{(it-1)}$ 是矩阵 \mathbf{A} 在 TALS 算法中第 $it-1$ 次迭代出的估计值， $\mathbf{A}^{(new)}$ 是第 it 次迭代的矩阵。 $(\mathbf{A}^{(it-1)} - \mathbf{A}^{(it-2)})$ 定义了循环的方向。矩阵 $\mathbf{B}^{(new)}$ 和 $\mathbf{C}^{(new)}$ 可以使用相同的因子 R_{LS} ，用同样的方法得到。 R_{LS} 是一个固定值，通常取 1.2、1.3 或者 $it^{1/3}$ 。

TALS-LS 算法的具体步骤如下：

步骤 1：初始化 $\mathbf{A}^{(it-2)}$ 、 $\mathbf{A}^{(it-1)}$ 、 $\mathbf{B}^{(it-2)}$ 、 $\mathbf{B}^{(it-1)}$ 、 $\mathbf{C}^{(it-2)}$ 、 $\mathbf{C}^{(it-1)}$ ，根据式 (3.4.10) 计算 $\gamma^{(it-1)}$ 。

步骤 2：通过以下公式计算 $\mathbf{A}^{(new)}$ 、 $\mathbf{B}^{(new)}$ 、 $\mathbf{C}^{(new)}$ 。

$$\mathbf{A}^{(new)} = \mathbf{A}^{(it-2)} + R_{LS}(\mathbf{A}^{(it-1)} - \mathbf{A}^{(it-2)}) \quad (3.4.13)$$

$$\mathbf{B}^{(new)} = \mathbf{B}^{(it-2)} + R_{LS}(\mathbf{B}^{(it-1)} - \mathbf{B}^{(it-2)}) \quad (3.4.14)$$

$$\mathbf{C}^{(new)} = \mathbf{C}^{(it-2)} + R_{LS}(\mathbf{C}^{(it-1)} - \mathbf{C}^{(it-2)}) \quad (3.4.15)$$

步骤 3：根据式 (3.4.10) 计算 $\gamma^{(new)}$ 。

如果 $\gamma^{(new)} < \gamma^{(it-1)}$ ，则令 $\mathbf{A}_0 = \mathbf{A}^{(new)}$ ， $\mathbf{B}_0 = \mathbf{B}^{(new)}$ ， $\mathbf{C}_0 = \mathbf{C}^{(new)}$ 。

如果 $\gamma^{(new)} \geq \gamma^{(it-1)}$ ，则令 $\mathbf{A}_0 = \mathbf{A}^{(it-1)}$ ， $\mathbf{B}_0 = \mathbf{B}^{(it-1)}$ ， $\mathbf{C}_0 = \mathbf{C}^{(it-1)}$ 。

步骤 4：用以下公式估计 $\mathbf{A}^{(it)}$ 、 $\mathbf{B}^{(it)}$ 、 $\mathbf{C}^{(it)}$ 。

$$\mathbf{A}^{(it)} = \mathbf{X}^{(I \times JK)} (\mathbf{Z}_a^+)^T, \text{ 其中 } \mathbf{Z}_a = \mathbf{C}_0 \odot \mathbf{B}_0 \quad (3.4.16)$$

$$\mathbf{B}^{(it)} = \mathbf{X}^{(J \times IK)} (\mathbf{Z}_b^+)^T, \text{ 其中 } \mathbf{Z}_b = \mathbf{A}^{(it)} \odot \mathbf{C}_0 \quad (3.4.17)$$

$$\mathbf{C}^{(\text{it})} = \mathbf{X}^{(K \times IJ)} (\mathbf{Z}_c^+)^T, \text{ 其中 } \mathbf{Z}_c = \mathbf{B}^{(\text{it})} \odot \mathbf{A}^{(\text{it})} \quad (3.4.18)$$

步骤 5: 根据式 (3.4.10) 计算 $\gamma^{(\text{it})}$ 。

如果 $|\gamma^{(\text{it})} - \gamma^{(\text{it}-1)}| > \varepsilon$, 则令 $\text{it} \leftarrow \text{it} + 1$, 并返回步骤 2。

如果 $|\gamma^{(\text{it})} - \gamma^{(\text{it}-1)}| \leq \varepsilon$, 则算法收敛。

3.4.3 四线性分解和四线性交替最小二乘

下面将三线性模型扩展成四线性模型。考虑以下四线性模型:

$$\mathbf{R}_{ij} = \begin{bmatrix} \mathbf{R}_{i1} \\ \mathbf{R}_{i2} \\ \vdots \\ \mathbf{R}_{iJ} \end{bmatrix} = \begin{bmatrix} \mathbf{SD}_1(\mathbf{G})\mathbf{D}_1(\mathbf{H}) \\ \mathbf{SD}_1(\mathbf{G})\mathbf{D}_2(\mathbf{H}) \\ \vdots \\ \mathbf{SD}_I(\mathbf{G})\mathbf{D}_J(\mathbf{H}) \end{bmatrix} \mathbf{B}^T \quad (3.4.19)$$

即

$$\mathbf{R}_{ij} = \mathbf{SD}_i(\mathbf{G})\mathbf{D}_j(\mathbf{H})\mathbf{B}^T, \quad i=1,2,\dots,I, \quad j=1,2,\dots,J \quad (3.4.20)$$

将 $I \times J$ 个切片积累成为 $I \times J \times N \times P$ 的四维数据集 \mathbf{R} , 则其四线性模型有以下形式:

$$r_{i,j,n,p} = \sum_{f=1}^K g_{i,f} h_{j,f} b_{n,f} s_{p,f}, \quad i=1,\dots,I, \quad j=1,\dots,J, \quad n=1,\dots,N, \quad p=1,\dots,P \quad (3.4.21)$$

其中, $s_{p,f}$ 是矩阵 \mathbf{S} 中 (p, f) 元素, $b_{n,f}$ 是矩阵 \mathbf{B} 中 (n, f) 的元素, 其他也是如此。式 (3.4.21) 称为四线性模型。

根据其对称性, 可得到其他三种形式:

$$\mathbf{U}_{j,p} = \mathbf{BD}_j(\mathbf{H})\mathbf{D}_p(\mathbf{S})\mathbf{G}^T, \quad j=1,\dots,J, \quad p=1,\dots,P \quad (3.4.22)$$

$$\mathbf{V}_{p,n} = \mathbf{GD}_p(\mathbf{S})\mathbf{D}_n(\mathbf{B})\mathbf{H}^T, \quad p=1,\dots,P, \quad n=1,\dots,N \quad (3.4.23)$$

$$\mathbf{W}_{n,i} = \mathbf{HD}_n(\mathbf{B})\mathbf{D}_i(\mathbf{G})\mathbf{S}^T, \quad n=1,\dots,N, \quad i=1,\dots,I \quad (3.4.24)$$

1. 四线性交替最小二乘算法

四线性交替最小二乘算法 (Quadrilinear Alternating Least Square, QALS) 是四线性模型数据检测的一种常用方法。QALS 的基本思想很简单, 就是每一步更新一个矩阵, 更新的办法是: 对余下的矩阵, 依据前一次估计的结果, 利用最小二乘法来更新; 继续对其他矩阵进行更新; 重复以上步骤直到算法收敛。

根据式 (3.4.19), 则其最小二乘拟合要求满足下式:

$$\min_{\mathbf{G}, \mathbf{H}, \mathbf{S}, \mathbf{B}} \left\| \begin{bmatrix} \mathbf{R}_{i1} \\ \mathbf{R}_{i2} \\ \vdots \\ \mathbf{R}_{iJ} \end{bmatrix} - \begin{bmatrix} \mathbf{SD}_1(\mathbf{G})\mathbf{D}_1(\mathbf{H}) \\ \mathbf{SD}_1(\mathbf{G})\mathbf{D}_2(\mathbf{H}) \\ \vdots \\ \mathbf{SD}_I(\mathbf{G})\mathbf{D}_J(\mathbf{H}) \end{bmatrix} \mathbf{B}^T \right\|_F \quad (3.4.25)$$

式中 \mathbf{R}_{ij} , $i=1,2,\dots,I$, $j=1,2,\dots,J$ 表示含噪声的切片, $\|\cdot\|_F$ 为 Frobenius 范数。最小二乘来更新 \mathbf{B} 矩阵:

$$\hat{\mathbf{B}}^T = \begin{bmatrix} \hat{\mathbf{S}}D_1(\hat{\mathbf{G}})D_1(\hat{\mathbf{H}}) \\ \hat{\mathbf{S}}D_1(\hat{\mathbf{G}})D_2(\hat{\mathbf{H}}) \\ \vdots \\ \hat{\mathbf{S}}D_I(\hat{\mathbf{G}})D_J(\hat{\mathbf{H}}) \end{bmatrix}^+ \begin{bmatrix} \tilde{\mathbf{R}}_{11} \\ \tilde{\mathbf{R}}_{12} \\ \vdots \\ \tilde{\mathbf{R}}_{IJ} \end{bmatrix} \quad (3.4.26)$$

同样利用四线性模型的第二种形式 $\mathbf{U}_{j,p} = \mathbf{B}D_j(\mathbf{H})D_p(\mathbf{S})\mathbf{G}^T$, $j=1,2,\dots,J$, $p=1,\dots,P$ 更新 \mathbf{G} 矩阵:

$$\hat{\mathbf{G}}^T = \begin{bmatrix} \hat{\mathbf{B}}D_1(\hat{\mathbf{H}})D_1(\hat{\mathbf{S}}) \\ \hat{\mathbf{B}}D_1(\hat{\mathbf{H}})D_2(\hat{\mathbf{S}}) \\ \vdots \\ \hat{\mathbf{B}}D_J(\hat{\mathbf{H}})D_P(\hat{\mathbf{S}}) \end{bmatrix}^+ \begin{bmatrix} \tilde{\mathbf{U}}_{11} \\ \tilde{\mathbf{U}}_{12} \\ \vdots \\ \tilde{\mathbf{U}}_{JP} \end{bmatrix} \quad (3.4.27)$$

利用四线性模型的第三种形式 $\mathbf{V}_{p,n} = \mathbf{G}D_p(\mathbf{S})D_n(\mathbf{B})\mathbf{H}^T$, $p=1,\dots,P$, $n=1,\dots,N$ 更新 \mathbf{H} 矩阵:

$$\hat{\mathbf{H}}^T = \begin{bmatrix} \hat{\mathbf{G}}D_1(\hat{\mathbf{S}})D_1(\hat{\mathbf{B}}) \\ \hat{\mathbf{G}}D_1(\hat{\mathbf{S}})D_2(\hat{\mathbf{B}}) \\ \vdots \\ \hat{\mathbf{G}}D_P(\hat{\mathbf{S}})D_N(\hat{\mathbf{B}}) \end{bmatrix}^+ \begin{bmatrix} \tilde{\mathbf{V}}_{11} \\ \tilde{\mathbf{V}}_{12} \\ \vdots \\ \tilde{\mathbf{V}}_{PN} \end{bmatrix} \quad (3.4.28)$$

利用四线性模型的第四种形式 $\mathbf{W}_{n,i} = \mathbf{H}D_n(\mathbf{B})D_i(\mathbf{G})\mathbf{S}^T$, $n=1,\dots,N$, $i=1,2,\dots,I$ 更新 \mathbf{S} 矩阵:

$$\hat{\mathbf{S}}^T = \begin{bmatrix} \hat{\mathbf{H}}D_1(\hat{\mathbf{B}})D_1(\hat{\mathbf{G}}) \\ \hat{\mathbf{H}}D_1(\hat{\mathbf{B}})D_2(\hat{\mathbf{G}}) \\ \vdots \\ \hat{\mathbf{H}}D_N(\hat{\mathbf{B}})D_I(\hat{\mathbf{G}}) \end{bmatrix}^+ \begin{bmatrix} \tilde{\mathbf{W}}_{11} \\ \tilde{\mathbf{W}}_{12} \\ \vdots \\ \tilde{\mathbf{W}}_{NI} \end{bmatrix} \quad (3.4.29)$$

2. 四线性分解模型算法的具体过程

步骤 1: 初始化方向矩阵 \mathbf{G} 、 \mathbf{H} 、 \mathbf{S} 和 \mathbf{B} 。

步骤 2: 根据式 (3.4.26) 更新 \mathbf{B} 矩阵。

步骤 3: 根据式 (3.4.27) 更新 \mathbf{G} 矩阵。

步骤 4: 根据式 (3.4.28) 更新 \mathbf{H} 矩阵。

步骤 5: 根据式 (3.4.29) 更新 \mathbf{S} 矩阵。

步骤 6: 重复步骤 2 到 5, 直到收敛。

3.4.4 基于正交约束 PARAFAC 分解

考虑一个三线性模型切片形式 \mathbf{X}_n :

$$\mathbf{X}_n = \mathbf{H}\mathbf{D}_n(\mathbf{S})\mathbf{G}^T + \mathbf{E}_X^{(n)}, \quad n=1, \dots, N$$

其中, $\mathbf{E}_X^{(n)}$ 表示时间方向第 n 个剖面的噪声。将 \mathbf{X}_n 沿时间顺序按行排列, 得到更加紧凑的形式:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_N \end{bmatrix} = \begin{bmatrix} \mathbf{H}\mathbf{D}_1(\mathbf{S})\mathbf{G}^T \\ \mathbf{H}\mathbf{D}_2(\mathbf{S})\mathbf{G}^T \\ \vdots \\ \mathbf{H}\mathbf{D}_N(\mathbf{S})\mathbf{G}^T \end{bmatrix} + \mathbf{E}_X = (\mathbf{S} \odot \mathbf{H})\mathbf{G}^T + \mathbf{E}_X \quad (3.4.30)$$

其中

$$\mathbf{E}_X = \begin{bmatrix} \mathbf{E}_X^{(1)} \\ \mathbf{E}_X^{(2)} \\ \vdots \\ \mathbf{E}_X^{(N)} \end{bmatrix}$$

根据三线性模型对称性, 得到其他切片:

$$\mathbf{Y}_p = \mathbf{S}\mathbf{D}_p(\mathbf{G})\mathbf{H}^T + \mathbf{E}_Y^{(p)}, \quad p=1, \dots, P$$

$$\mathbf{Z}_l = \mathbf{G}\mathbf{D}_l(\mathbf{H})\mathbf{S}^T + \mathbf{E}_Z^{(l)}, \quad l=1, \dots, L$$

构建

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_P \end{bmatrix} = \begin{bmatrix} \mathbf{S}\mathbf{D}_1(\mathbf{G})\mathbf{H}^T \\ \mathbf{S}\mathbf{D}_2(\mathbf{G})\mathbf{H}^T \\ \vdots \\ \mathbf{S}\mathbf{D}_P(\mathbf{G})\mathbf{H}^T \end{bmatrix} + \mathbf{E}_Y = (\mathbf{G} \odot \mathbf{S})\mathbf{H}^T + \mathbf{E}_Y \quad (3.4.31)$$

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \\ \vdots \\ \mathbf{Z}_L \end{bmatrix} = \begin{bmatrix} \mathbf{G}\mathbf{D}_1(\mathbf{H})\mathbf{S}^T \\ \mathbf{G}\mathbf{D}_2(\mathbf{H})\mathbf{S}^T \\ \vdots \\ \mathbf{G}\mathbf{D}_L(\mathbf{H})\mathbf{S}^T \end{bmatrix} + \mathbf{E}_Z = (\mathbf{H} \odot \mathbf{G})\mathbf{S}^T + \mathbf{E}_Z \quad (3.4.32)$$

其中, $\mathbf{E}_Y, \mathbf{E}_Z$ 分别是噪声矩阵。注意到, 与剖面矩阵 $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ 类似, $\mathbf{E}_X, \mathbf{E}_Y, \mathbf{E}_Z$ 具有相同的元素和统计特性, 它们仅在元素的排列顺序上有所不同。

在实际应用过程中, 如果针对不同的数据结构对 PARAFAC 模型中的承载矩阵进行合理约束, 可以提高算法的拟合精度和收敛速度。因此, 本节将承载矩阵的正交结构融入 PARAFAC 分解, 文献[1]提出了一种更适用于处理 DS-CDMA 信号的正交约束 PARAFAC (Orthogonal Constrained PARAFAC, OC-PARAFAC) 接收机, 其中利用了扩频码正交性。

该接收机使用正交约束三线性迭代最小二乘（Orthogonal Constrained Trilinear Alternative Least Square, OCTALS）方法对接收信号进行处理，完成多用户检测工作。下面介绍 OCTALS 算法。

OCTALS 算法是 TALS 算法的拓展，它将承载矩阵 \mathbf{G} 的正交结构以约束的形式应用到 TALS 中，在迭代过程中对 \mathbf{G} 矩阵进行重构，迭代后得到的承载矩阵的估计值 $\hat{\mathbf{G}}$ 的列向量相互正交。承载矩阵 \mathbf{G} 的正交重构过程如下。

根据式 (3.4.30)，令 $\mathbf{F} = \mathbf{S} \odot \mathbf{H}$ ，问题转化为：在 $\mathbf{G}^T \mathbf{G} = \mathbf{I}$ 的约束条件下求取 \mathbf{G} ，使得 $\|\mathbf{X} - \mathbf{F}\mathbf{G}^T\|_{\text{F}}^2$ 最小，即

$$\hat{\mathbf{G}} = \arg \min_{\mathbf{G}^T \mathbf{G} = \mathbf{I}} \|\mathbf{X} - \mathbf{F}\mathbf{G}^T\|_{\text{F}}^2 \quad (3.4.33)$$

这是一个约束最小二乘问题，可以采用 Lagrange 乘子法对式 (3.4.33) 进行求解，设 λ 为一常数对称矩阵，此约束最小二乘问题可以表示为下列形式：

$$\begin{aligned} f(\mathbf{G}, \lambda) &= \|\mathbf{X} - \mathbf{F}\mathbf{G}^T\|_{\text{F}}^2 + \lambda(\mathbf{G}^T \mathbf{G} - \mathbf{I}) \\ &= \text{tr}[(\mathbf{X} - \mathbf{F}\mathbf{G}^T)^{\text{H}}(\mathbf{X} - \mathbf{F}\mathbf{G}^T)] + \lambda(\mathbf{G}^T \mathbf{G} - \mathbf{I}) \end{aligned} \quad (3.4.34)$$

对 \mathbf{G} 进行求导，使其为 0，有

$$\frac{\partial f(\mathbf{G}, \lambda)}{\partial \mathbf{G}} = \mathbf{G} \text{Re}(\mathbf{F}^{\text{H}} \mathbf{F}) - \text{Re}(\mathbf{X}^{\text{H}} \mathbf{F}) + \mathbf{G} \lambda = 0 \quad (3.4.35)$$

得到矩阵 \mathbf{G} 的值为

$$\mathbf{G} = \text{Re}(\mathbf{X}^{\text{H}} \mathbf{F})(\lambda + \text{Re}(\mathbf{F}^{\text{H}} \mathbf{F}))^{-1} \quad (3.4.36)$$

将式 (3.4.36) 代入约束条件，即

$$(\text{Re}(\mathbf{X}^{\text{H}} \mathbf{F})(\lambda + \text{Re}(\mathbf{F}^{\text{H}} \mathbf{F}))^{-1})(\text{Re}(\mathbf{X}^{\text{H}} \mathbf{F})(\lambda + \text{Re}(\mathbf{F}^{\text{H}} \mathbf{F}))^{-1}) = 0 \quad (3.4.37)$$

则式 (3.4.37) 改写为

$$(\lambda + \text{Re}(\mathbf{F}^{\text{H}} \mathbf{F}))^{-1} = (\mathbf{U}^T \mathbf{U})^{-0.5} \quad (3.4.38)$$

其中

$$\mathbf{U} = \text{Re}(\mathbf{X}^{\text{H}} \mathbf{F})$$

\mathbf{G} 的估计值为

$$\hat{\mathbf{G}} = \mathbf{U}(\mathbf{U}^T \mathbf{U})^{-0.5} \quad (3.4.39)$$

式 (3.4.39) 得到的估计值 $\hat{\mathbf{G}}$ 是一个正交矩阵。至此， \mathbf{G} 矩阵的正交重构完成。

下面，对 \mathbf{H} 和 \mathbf{S} 进行求解。根据式 (3.4.31)，利用最小二乘准则得到 \mathbf{H} 的估计值为

$$\begin{aligned} \hat{\mathbf{H}}^T &= \arg \min_{\mathbf{H}} \|\mathbf{Y} - (\mathbf{G} \odot \mathbf{S})\mathbf{H}^T\|_{\text{F}}^2 \\ &= (\mathbf{G} \odot \mathbf{S})^{\dagger} \mathbf{Y} \end{aligned} \quad (3.4.40)$$

同样，根据式 (3.4.32)，利用最小二乘准则， \mathbf{S} 的估计值为

$$\begin{aligned} \hat{\mathbf{S}}^T &= \arg \min_{\mathbf{S}} \|\mathbf{Z} - (\mathbf{H} \odot \mathbf{G})\mathbf{S}^T\|_{\text{F}}^2 \\ &= (\mathbf{H} \odot \mathbf{G})^{\dagger} \mathbf{Z} \end{aligned} \quad (3.4.41)$$

OCTALS 算法的基本思想是：利用式 (3.4.39)、式 (3.4.40) 和式 (3.4.41)，每次更新

\mathbf{G} 、 \mathbf{S} 、 \mathbf{H} 中的一个矩阵，将更新后的矩阵作为下一个矩阵更新时的参数，重复以上步骤直至算法收敛。记 $(\bullet)^{(k)}$ 表示第 k 次迭代的矩阵，下面给出算法的具体实现。

步骤 1: 使用随机值初始化承载矩阵 $\hat{\mathbf{G}}^{(0)}$ 、 $\hat{\mathbf{S}}^{(0)}$ 、 $\hat{\mathbf{H}}^{(0)}$ ， $k=0$ 。

步骤 2: $k=k+1$ 。

步骤 3: 利用式 (3.4.39) 及 \mathbf{X} 、 $\hat{\mathbf{S}}^{(k-1)}$ 、 $\hat{\mathbf{H}}^{(k-1)}$ ，计算 \mathbf{G} 的第 k 次迭代值 $\hat{\mathbf{G}}^{(k)}$ 。

步骤 4: 利用式 (3.4.40) 及 \mathbf{Y} 、 $\hat{\mathbf{G}}^{(k)}$ 、 $\hat{\mathbf{S}}^{(k-1)}$ ，计算 \mathbf{H} 的第 k 次迭代值 $\hat{\mathbf{H}}^{(k)}$ 。

步骤 5: 利用式 (3.4.41) 及 \mathbf{Z} 、 $\hat{\mathbf{G}}^{(k)}$ 、 $\hat{\mathbf{H}}^{(k)}$ ，计算 \mathbf{S} 的第 k 次迭代值 $\hat{\mathbf{S}}^{(k)}$ 。

步骤 6: 重复步骤 2~5，令

$$\varepsilon = \|\mathbf{X} - [(\hat{\mathbf{S}}^{(k-1)} \odot \hat{\mathbf{H}}^{(k-1)})] \hat{\mathbf{G}}^{(k-1)\top}\|_{\text{F}}^2 - \|\mathbf{X} - [(\hat{\mathbf{S}}^{(k)} \odot \hat{\mathbf{H}}^{(k)})] \hat{\mathbf{G}}^{(k)\top}\|_{\text{F}}^2$$

当 ε 小于一个很小的数时，迭代结束。

算法结束后，得到估计矩阵 $\hat{\mathbf{G}}, \hat{\mathbf{S}}, \hat{\mathbf{H}}$ ，其中， $\hat{\mathbf{G}}$ 为正交矩阵。仿真结果表明， $\hat{\mathbf{G}}$ 经重构后，更逼近于真实的 \mathbf{G} ，参数估计精度得到改善，同时提高了迭代收敛过程的速度和稳定性。

3.4.5 结构约束 PARAFAC 分解

1. 结构约束 PARAFAC 模型

首先，给出 PARAFAC 模型的表达式。令 \mathbf{X} 是 $I \times J \times K$ 的三维数据矩阵， $x_{ijk}, i=1, \dots, I, j=1, \dots, J, k=1, \dots, K$ 是 \mathbf{X} 中的元素， $\mathbf{A} \in \mathbb{C}^{I \times F}$ ， $\mathbf{B} \in \mathbb{C}^{J \times F}$ ， $\mathbf{C} \in \mathbb{C}^{K \times F}$ 是 \mathbf{X} 的三个承载矩阵，其元素为 $a_{ir}, b_{jr}, c_{kr}, r=1, \dots, R$ ， R 是因子数目，PARAFAC 模型的标量形式为

$$x_{ijk} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr} \quad (3.4.42)$$

其沿 \mathbf{X} 轴的切面形式为

$$\mathbf{X}^{IJ \times K} = \begin{bmatrix} \mathbf{B}\mathbf{D}_1(\mathbf{A})\mathbf{C}^{\text{T}} \\ \mathbf{B}\mathbf{D}_2(\mathbf{A})\mathbf{C}^{\text{T}} \\ \vdots \\ \mathbf{B}\mathbf{D}_I(\mathbf{A})\mathbf{C}^{\text{T}} \end{bmatrix} = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^{\text{T}} \quad (3.4.43)$$

在通信领域，信号经调制后会具有一定的结构性质，典型的有以下几种。

① 恒模 (Constant Modulus, CM): 信号的模值恒定，如调频信号、调相信号、 M 元相移键控信号等。

② 有限字符 (Finite Alphabet, FA): 信号在一个有限字符集中取。

诸多数字调制信号都属于此类，如正交幅度调制信号、脉冲幅度调制信号等。

通常，这些结构在接收端是已知的。然而，在上述应用中，算法在进行信号处理时仅仅把信源矩阵简单地当作一个普通的数据矩阵进行处理，而忽略了信源本身的结构性质。

因此, 针对上述问题, 本节通过建立结构约束 PARAFAC 模型, 将信源的结构性质融入 PARAFAC 分析过程中。假设 \mathbf{C} 矩阵为信源矩阵, 结构约束 PARAFAC 模型的标量形式可以表示为

$$\underline{x}_{ijk} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr}, \quad c_{kr} \in \Omega$$

其中, Ω 是具有一定结构特征的数据集合, 本节中 Ω 指恒模数集或有限字符数集。经研究表明, 将信源矩阵的结构性质融入 PARAFAC 分析, 不仅可以改善 PARAFAC 模型的分解唯一性条件, 而且可以提高信号处理算法的性能。首先, 讨论结构约束 PARAFAC 分解唯一性。

2. 结构约束 PARAFAC 唯一性条件

当 PARAFAC 模型的承载矩阵具有一定结构时, 其分解唯一性条件会发生变化, 它不仅取决于普通 PARAFAC 模型的唯一性条件, 而且依赖于融入的结构特征。普通 PARAFAC 模型的唯一性条件是

$$k_A + k_B + k_C \geq 2R + 2 \quad (3.4.44)$$

其中, k_A 表示矩阵 \mathbf{A} 的 k -秩。由于承载矩阵 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 分别为 $I \times R, J \times R, K \times R$ 的矩阵, 根据 k -秩的定义, 有

$$k_A \leq R, k_B \leq R, k_C \leq R \quad (3.4.45)$$

式 (3.4.44) 和式 (3.4.45) 表明, 若实现 PARAFAC 模型分解唯一, 必须满足下列条件:

$$\min(k_A + k_B + k_C) \geq 2 \quad (3.4.46)$$

根据 k -秩的定义, 若某一矩阵中含有线性相关列, 则该矩阵的 k -秩为 1, 式 (3.4.46) 不再满足。因此, PARAFAC 模型分解唯一的一个必要条件为其三个承载矩阵中均不可以含有任何相关列, 这个条件在一定程度上限制了 PARAFAC 分析在通信中的应用。然而, 将结构约束融入 PARAFAC 模型以后, 模型分解的唯一性条件得到改善。以恒模结构为例, 假设信源矩阵 \mathbf{C} 具有恒模性质, 考虑式 (3.4.42) 给出的 PARAFAC 模型沿 X 轴的切面形式, 令 $\mathbf{D} = \mathbf{A} \odot \mathbf{B}$, 式 (3.4.43) 转化为 $\mathbf{X} = \mathbf{D}\mathbf{C}^T$ 。考虑参考文献[11]给出的恒模约束矩阵的分解唯一性定理。

定理 3.4.1 (恒模分解唯一) [11]:

考虑数据模型:

$$\mathbf{X} = \mathbf{D}\mathbf{C}^T \quad (3.4.47)$$

其中, \mathbf{D} 是 $IJ \times R$ 的满列秩矩阵, \mathbf{C} 是 $K \times R$ 恒模矩阵, 即 \mathbf{C} 中的元素 $\|c_{kr}\| = 1, k = 1, \dots, K, r = 1, \dots, R$ 。如果

$$K \geq R(R-1) - 1 \quad (3.4.48)$$

则式 (3.4.47) 分解唯一, 即若存在另一组矩阵 $\bar{\mathbf{D}} \in \mathbb{C}^{IJ \times R}$ 、 $\bar{\mathbf{C}} \in \mathbb{C}^{K \times R}$ 满足 $\mathbf{X} = \bar{\mathbf{D}}\bar{\mathbf{C}}^T$, 且 $\|\bar{c}_{kr}\| = 1, k = 1, \dots, K, r = 1, \dots, R$, 则有

$$\bar{\mathbf{D}} = \mathbf{D}\Pi\mathbf{A}_1, \quad \bar{\mathbf{C}} = \mathbf{C}\Pi\mathbf{A}_2 \quad (3.4.49)$$

其中, Π 是列交换矩阵, $\mathbf{A}_{1,2}$ 是尺度模糊矩阵。

定理 3.4.1 表明, 二维矩阵的恒模分解可以实现分解唯一。由于 $\mathbf{D} = \mathbf{A} \odot \mathbf{B}$, 下面将说明, 在满足定理 3.4.1 的情况下, 若 \mathbf{D} 分解唯一, 则 \mathbf{A}, \mathbf{B} 亦分解唯一。记 $\bar{\mathbf{d}}_r$ 是 $\bar{\mathbf{D}}$ 的第 r 列, $\mathbf{d}_r, \mathbf{a}_r, \mathbf{b}_r$ 分别为 $\mathbf{D}, \mathbf{A}, \mathbf{B}$ 的第 r 列, 由于

$$\bar{\mathbf{D}} = \mathbf{D}\Pi\mathbf{A} \quad (3.4.50)$$

式 (3.4.50) 的向量形式为

$$[\bar{\mathbf{d}}_1, \dots, \bar{\mathbf{d}}_R] = [\mathbf{d}_1, \dots, \mathbf{d}_R] \Pi \mathbf{A} \quad (3.4.51)$$

以向量为单位进行处理, 忽略列交换模糊, 有

$$\bar{\mathbf{d}}_r = \lambda \mathbf{d}_r = \lambda(\mathbf{a}_r \otimes \mathbf{b}_r) \quad (3.4.52)$$

将 $\bar{\mathbf{d}}_r$ 重构为 $I \times J$ 的矩阵:

$$\bar{\mathbf{D}}_r = \text{unvec}(\bar{\mathbf{d}}_r, J, I) = \lambda \text{unvec}(\mathbf{a}_r \otimes \mathbf{b}_r) = \lambda \mathbf{b}_r^T \mathbf{a}_r \quad (3.4.53)$$

从式 (3.4.53) 可知, $\bar{\mathbf{D}}_r$ 是一个秩为 1 的矩阵。记 $\bar{\mathbf{a}}_r, \bar{\mathbf{b}}_r$ 分别为 \mathbf{D}_r 的非零奇异值对应的左向量和右向量, 有

$$\bar{\mathbf{a}}_r = \lambda \beta \mathbf{a}_r, \quad \bar{\mathbf{b}}_r = \beta^{-1} \mathbf{b}_r \quad (3.4.54)$$

其中 β 为非零系数。记 $\bar{\mathbf{A}} = [\bar{\mathbf{a}}_1, \dots, \bar{\mathbf{a}}_R]$, $\bar{\mathbf{B}} = [\bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_R]$, 有

$$\bar{\mathbf{A}} = [\bar{\mathbf{a}}_1, \dots, \bar{\mathbf{a}}_R] = [\mathbf{a}_1, \dots, \mathbf{a}_R] \begin{bmatrix} \lambda \beta_1 & & \\ & \ddots & \\ & & \lambda \beta_R \end{bmatrix} = \lambda \mathbf{A} \mathbf{A} \quad (3.4.55)$$

$$\bar{\mathbf{B}} = [\bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_R] = [\mathbf{b}_1, \dots, \mathbf{b}_R] \begin{bmatrix} \beta_1^{-1} & & \\ & \ddots & \\ & & \beta_R^{-1} \end{bmatrix} = \mathbf{B} \mathbf{A}^{-1} \quad (3.4.56)$$

其中

$$\mathbf{A} = \begin{bmatrix} \beta_1 & & \\ & \ddots & \\ & & \beta_R \end{bmatrix} \quad (3.4.57)$$

根据式 (3.4.55) 和式 (3.4.56) 可知, 由 $\bar{\mathbf{D}}$ 得到的 $\bar{\mathbf{A}}, \bar{\mathbf{B}}$ 分别与 \mathbf{A}, \mathbf{B} 本质相等, 因此, 若 \mathbf{D} 分解唯一, 则 \mathbf{A}, \mathbf{B} 亦分解唯一。

在满足定理 3.4.1 的情况下, 恒模约束 PARAFAC 模型分解唯一。在定理 3.4.1 中, 要求矩阵 \mathbf{D} 列满秩。由于 \mathbf{D} 是 \mathbf{A}, \mathbf{B} 的 Khatri-Rao 积, 恒模约束 PARAFAC 模型的分解唯一性问题转化为: \mathbf{A}, \mathbf{B} 满足什么条件时, 矩阵 \mathbf{D} 达到列满秩。这是一个 Khatri-Rao 积的列满秩判定问题, 已有学者对此进行了深入的研究, 下面给出其中使用较为广泛的判定条件。

定理 3.4.2 (Khatri-Rao 积列满秩判定定理) ^[12]:

考虑

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1, \dots, \mathbf{a}_R \otimes \mathbf{b}_R] \quad (3.4.58)$$

其中, $\mathbf{A} \in \mathbb{C}^{I \times R}$, $\mathbf{B} \in \mathbb{C}^{J \times R}$, $\mathbf{a}_r, \mathbf{b}_r, r=1, \dots, R$ 分别是 \mathbf{A} 和 \mathbf{B} 的列向量, 若

$$k_A + k_B \geq R + 1 \quad (3.4.59)$$

则 $\mathbf{A} \odot \mathbf{B}$ 满列秩。

定理 3.4.2 指出, 若 \mathbf{A}, \mathbf{B} 中有一个矩阵 (这里假设是 \mathbf{A}) 满 k -秩, 则 \mathbf{B} 的 k -秩的最小值为 1。这说明, 即使 \mathbf{B} 矩阵中存在线性相关列 (此时 k -秩为 1), $\mathbf{A} \odot \mathbf{B}$ 依然可以达到列满秩, 可以满足定理 3.4.2 的条件, 此时, 式 (3.4.47) 可以实现恒模分解唯一, 恒模约束 PARAFAC 分解同样可以实现唯一。

需要指出, 定理 3.4.2 给出的 Khatri-Rao 积列满秩判定条件仅为充分条件, 在一些情况下, Khatri-Rao 积列满秩条件会比定理 3.4.2 弱, 下面给出另一个判定定理。

定理 3.4.3: 考虑矩阵 $\mathbf{A} \in \mathbb{C}^{I \times R}$, $\mathbf{B} \in \mathbb{C}^{J \times R}$, 有

$$r_{\mathbf{A} \odot \mathbf{B}} = k_{\mathbf{A} \odot \mathbf{B}} = \min(IJ, R), P_{\zeta}(\mathbf{C}^{(I+J)R}) - a.s \quad (3.4.60)$$

其中, $P_{\zeta}(\mathbf{C}^{(I+J)R}) - a.s$ 表示 \mathbf{A}, \mathbf{B} 中的元素属于 $(I+J)R$ 维连续拉贝尔集。

定理 3.4.3 给出的 Khatri-Rao 积列满秩判定条件要弱于定理 3.4.2, 当 \mathbf{A}, \mathbf{B} 中的元素满足 $(I+J)R$ 维连续拉贝尔集时, 仅需 $IJ \geq R$ 即可满足 $\mathbf{A} \odot \mathbf{B}$ 列满秩。结合定理 3.4.1 和定理 3.4.2, 可以得到恒模约束 PARAFAC 分解唯一性定理。

定理 3.4.4 (恒模约束 PARAFAC 分解唯一性定理):

考虑平行因子模型:

$$\mathbf{X} = (\mathbf{A} \odot \mathbf{B}) \mathbf{C}^T \quad (3.4.61)$$

$\mathbf{A} \in \mathbb{C}^{I \times R}$ 、 $\mathbf{B} \in \mathbb{C}^{J \times R}$ 、 $\mathbf{C} \in \mathbb{C}^{K \times R}$ 分别为 \mathbf{X} 的三个承载矩阵, 其中, 矩阵 \mathbf{C} 列满秩且其元素具有恒模性质, 即 $\|\mathbf{c}_{kr}\| = 1, k=1, \dots, K, r=1, \dots, R$, 若下列条件满足:

$$\begin{cases} k_A + k_B \geq R + 1 \\ K \geq R(R-1) - 1 \end{cases} \quad (3.4.62)$$

则矩阵 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 分解唯一。

定理 3.4.4 的结论可以直接由定理 3.4.1 和定理 3.4.2 得到, 这里不再证明。定理 3.4.4 表明, 当 \mathbf{A}, \mathbf{B} 中含有线性相关列时, 恒模约束 PARAFAC 模型依然可以实现分解唯一。同样, 当数据矩阵 \mathbf{C} 满足有限字符性质时, 可以得到类似于定理 3.4.4 的结论, 这里不再赘述。

3. 基于结构约束 PARAFAC 分解算法

传统的基于 PARAFAC 模型的信号处理算法通常使用三线性交替最小二乘 (TALS) 对接收信号进行拟合。为了对结构约束 PARAFAC 模型进行拟合, 本小节将结构约束引入承载矩阵的迭代过程, 介绍两种新的拟合算法: 投影三线性最小二乘算法 (TALSP) 和连续干扰抵消三线性最小二乘算法 (TALSSIC)。TALSP 算法和 TALSSIC 算法均是块迭代算法, 它们在 \mathbf{C} 矩阵的求解方法上有所不同, TALSP 算法使用简单的结构映射实现 \mathbf{C} 矩阵的重构, 而 TALSSIC 算法使用连续干扰抵消策略对 \mathbf{C} 进行重构。利用上述两种算法可以设计基于结构约束 PARAFAC 模型的估计器, 用于信号检测和参数估计。下面以恒模约束为例介

绍 TALSP 和 TALSSIC 的实现。首先, 考虑接收端的加性噪声, 给出 PARAFAC 模型在三个方向的剖面形式:

$$\mathbf{X} = (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T + \mathbf{E}_X \quad (3.4.63)$$

$$\mathbf{Y} = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T + \mathbf{E}_Y \quad (3.4.64)$$

$$\mathbf{Z} = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T + \mathbf{E}_Z \quad (3.4.65)$$

$\mathbf{E}_X, \mathbf{E}_Y, \mathbf{E}_Z$ 是三个切面的噪声矩阵。算法的三个等价的代价函数为

$$\begin{aligned} f(\mathbf{A}, \mathbf{B}, \mathbf{C}; \mathbf{X}, \mathbf{Y}, \mathbf{Z}) &= \|\mathbf{X} - (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T\|_F \\ &= \|\mathbf{Y} - (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T\|_F \\ &= \|\mathbf{Z} - (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T\|_F \end{aligned} \quad (3.4.66)$$

(1) TALSP 算法

TALSP 算法使用最小二乘准则对三个模式矩阵 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 进行迭代求解。与传统的 TALS 算法不同, 在求解矩阵 \mathbf{C} 的过程中, TALSP 算法分两步对其进行求解, 首先, 根据式 (3.4.63), 求解 \mathbf{C} 矩阵的无约束最小二乘解:

$$\begin{aligned} \tilde{\mathbf{C}}^T &= \arg \min_{\mathbf{C}} \|\mathbf{X} - (\mathbf{A} \odot \mathbf{B})\mathbf{C}^T\|_F \\ &= (\mathbf{A} \odot \mathbf{B})^\dagger \mathbf{X} \end{aligned} \quad (3.4.67)$$

其次, 将 $\tilde{\mathbf{C}}$ 中每一个元素投影到恒模数集 Ω 中, 即

$$\hat{\mathbf{c}}_{kr} = \text{proj}_{\text{CM}}(\tilde{\mathbf{c}}_{kr}) \quad (3.4.68)$$

其中, $\text{proj}_{\text{CM}}(\bullet)$ 是恒模映射算子, $\text{proj}_{\text{CM}}(a) = a / \|a\|_2$ 。 \mathbf{C} 矩阵更新完毕, 此时, $\hat{\mathbf{C}}$ 中的元素的模恒定为 1。

根据式 (3.4.64), 求解 \mathbf{A} 矩阵的最小二乘解:

$$\begin{aligned} \hat{\mathbf{A}}^T &= \arg \min_{\mathbf{A}} \|\mathbf{Y} - (\mathbf{B} \odot \mathbf{C})\mathbf{A}^T\|_F \\ &= (\mathbf{B} \odot \mathbf{C})^\dagger \mathbf{Y} \end{aligned} \quad (3.4.69)$$

同样, 根据式 (3.4.65), 求解 \mathbf{B} 矩阵的最小二乘解:

$$\begin{aligned} \hat{\mathbf{B}}^T &= \arg \min_{\mathbf{B}} \|\mathbf{Z} - (\mathbf{C} \odot \mathbf{A})\mathbf{B}^T\|_F \\ &= (\mathbf{C} \odot \mathbf{A})^\dagger \mathbf{Z} \end{aligned} \quad (3.4.70)$$

利用上述方法, 每次更新一个矩阵, 本次更新的矩阵立刻参与下一个矩阵的求解, 记 $(\bullet)^{(k)}$ 表示第 k 次更新得到的矩阵, 下面给出 TALSP 算法的实现。

步骤 1: 初始化承载矩阵 $\hat{\mathbf{A}}^{(0)}$ 、 $\hat{\mathbf{B}}^{(0)}$ 、 $\hat{\mathbf{C}}^{(0)}$, $k=0$ 。

步骤 2: $k = k + 1$ 。

步骤 3: 利用式 (3.4.67), 式 (3.4.68) 及 \mathbf{X} 、 $\hat{\mathbf{A}}^{(k-1)}$ 、 $\hat{\mathbf{B}}^{(k-1)}$, 计算 \mathbf{C} 的第 k 次迭代值 $\hat{\mathbf{C}}^{(k)}$ 。

步骤 4: 利用式 (3.4.69) 及 \mathbf{Y} 、 $\hat{\mathbf{B}}^{(k-1)}$ 、 $\hat{\mathbf{C}}^{(k)}$, 计算 \mathbf{A} 的第 k 次迭代值 $\hat{\mathbf{A}}^{(k)}$ 。

步骤 5: 利用式 (3.4.70) 及 \mathbf{Z} 、 $\hat{\mathbf{C}}^{(k)}$ 、 $\hat{\mathbf{A}}^{(k)}$, 计算 \mathbf{B} 的第 k 次迭代值 $\hat{\mathbf{B}}^{(k)}$ 。

步骤 6: 重复步骤 2~5, 令

$$\varepsilon = \|X - (\hat{A}^{(k-1)} \odot \hat{B}^{(k-1)})\hat{C}^{(k-1)\top}\|_F^2 - \|X - (\hat{A}^{(k)} \odot \hat{B}^{(k)})\hat{C}^{(k)\top}\|_F^2$$

当 ε 小于一个很小的数时, 迭代结束。

在 \mathbf{C} 的求解过程中, 恒模映射使得每次迭代得到的 $\hat{\mathbf{C}}$ 为恒模矩阵。算法收敛后, 在满足定理 3.4.4 (式 3.4.61) 的情况下, 可以唯一地得到估计值 $\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}$, 实现信号检测和参数估计。

(2) TALSSIC 算法

TALSP 算法虽然形式简单, 复杂度较低, 但是算法在执行的过程中无法保证其每一次迭代都可以使得代价函数式 (3.4.66) 下降或保持, 恒模映射的存在破坏了 TALSP 算法的单调收敛性 (虽然在后面的仿真中, 发现 TALSP 算法在大部分情况下可以很快地收敛到正确的值)。为了解决 TALSP 算法的不收敛问题, 本节研究了 TALSSIC 算法, TALSSIC 算法仅在 \mathbf{C} 矩阵的求解环节与 TALSP 算法不同, 它将信号处理算法中常用的连续干扰抵消策略用于 \mathbf{C} 矩阵的重构, 此时, \mathbf{C} 矩阵的更新环节仅会使得代价函数下降或保持, 但不会使其上升, TALSSIC 算法是单调收敛的。下面给出 TALSSIC 算法的思想和实现, 同时说明 TALSSIC 算法的单调收敛性。

根据式 (3.4.63), 假设第 k 次迭代得到的承载矩阵的估计值为 $\hat{\mathbf{A}}^{(k)}, \hat{\mathbf{B}}^{(k)}, \hat{\mathbf{C}}^{(k)}$, 令 $\hat{\mathbf{D}}^{(k)} = \hat{\mathbf{A}}^{(k)} \odot \hat{\mathbf{B}}^{(k)}$, $\hat{\mathbf{d}}_r^{(k)}, \hat{\mathbf{a}}_r^{(k)}, \hat{\mathbf{b}}_r^{(k)}, \hat{\mathbf{c}}_r^{(k)}$ 分别表示 $\hat{\mathbf{D}}^{(k)}, \hat{\mathbf{A}}^{(k)}, \hat{\mathbf{B}}^{(k)}, \hat{\mathbf{C}}^{(k)}$ 的第 r 列。根据式 (3.4.47), 有

$$\mathbf{X} = \hat{\mathbf{D}}^{(k)}\hat{\mathbf{C}}^{(k)} + \mathbf{E}_1^{(k+1)} = \sum_{r=1}^R \hat{\mathbf{d}}_r^{(k)}\hat{\mathbf{c}}_r^{(k)\top} + \mathbf{E}_1^{(k+1)} \quad (3.4.71)$$

$\mathbf{E}_1^{(k+1)}$ 是第 $k+1$ 次迭代中求解 $\hat{\mathbf{c}}_1$ 的噪声矩阵。令

$$\mathbf{X}_1 = \sum_{r=2}^R \hat{\mathbf{d}}_r^{(k)}\hat{\mathbf{c}}_r^{(k)\top} \quad (3.4.72)$$

则

$$\mathbf{X} = \mathbf{X}_1 + \hat{\mathbf{d}}_1^{(k)}\hat{\mathbf{c}}_1^{(k)\top} + \mathbf{E}_1^{(k+1)} \quad (3.4.73)$$

利用最小二乘准则, \mathbf{c}_1 的第 $k+1$ 次迭代值为

$$\begin{aligned} \tilde{\mathbf{c}}_1^{(k+1)} &= \arg \min_{\mathbf{c}_1^{(k)}} \|\mathbf{X} - \mathbf{X}_1 - \hat{\mathbf{d}}_1^{(k)}\hat{\mathbf{c}}_1^{(k)\top}\|_F^2 \\ &= \left(\frac{(\hat{\mathbf{d}}_1^{(k)})^H}{\|\hat{\mathbf{d}}_1^{(k)}\|_2^2} (\mathbf{X} - \mathbf{X}_1) \right)^T \end{aligned} \quad (3.4.74)$$

对向量 $\tilde{\mathbf{c}}_1^{(k+1)}$ 中的元素进行恒模映射:

$$\hat{\mathbf{c}}_1^{(k+1)} = \text{proj}_{\text{CM}}(\tilde{\mathbf{c}}_1^{(k+1)}) \quad (3.4.75)$$

将 $\hat{\mathbf{c}}_1^{(k+1)}$ 更新至 $\hat{\mathbf{C}}^{(k)}$ 并参与 $\hat{\mathbf{c}}_2^{(k+1)}$ 的求解。令

$$\mathbf{X}_2 = \sum_{r=3}^R \hat{\mathbf{d}}_r^{(k)}\hat{\mathbf{c}}_r^{(k)\top} + \hat{\mathbf{d}}_1^{(k)}\hat{\mathbf{c}}_1^{(k+1)\top} \quad (3.4.76)$$

$$\mathbf{X} = \mathbf{X}_2 + \hat{\mathbf{d}}_2^{(k)}\hat{\mathbf{c}}_2^{(k)\top} + \mathbf{E}_2^{(k+1)}$$

$\mathbf{E}_2^{(k+1)}$ 是第 $k+1$ 次迭代中求解 $\hat{\mathbf{c}}_2$ 的噪声矩阵。利用最小二乘准则, \mathbf{c}_2 的第 $k+1$ 次迭代值为:

$$\begin{aligned}\tilde{\mathbf{c}}_2^{(k+1)} &= \arg \min_{\mathbf{c}_2^{(k)}} \|\mathbf{X} - \mathbf{X}_2 - \hat{\mathbf{d}}_2^{(k)} \hat{\mathbf{c}}_2^{(k)\top}\|_{\text{F}}^2 \\ &= \left(\frac{(\hat{\mathbf{d}}_2^{(k)})^{\text{H}}}{\|\hat{\mathbf{d}}_2^{(k)}\|_2^2} (\mathbf{X} - \mathbf{X}_2) \right)^{\text{T}}\end{aligned}\quad (3.4.77)$$

同样，对向量 $\tilde{\mathbf{c}}_2^{(k+1)}$ 中的元素进行恒模映射：

$$\hat{\mathbf{c}}_2^{(k+1)} = \text{proj}_{\text{CM}}(\tilde{\mathbf{c}}_2^{(k+1)}) \quad (3.4.78)$$

以此类推，令

$$\begin{aligned}\mathbf{X}_i &= \sum_{r=i+1}^R \hat{\mathbf{d}}_r^{(k)} \hat{\mathbf{c}}_r^{(k)\top} + \sum_{r=1}^{i-1} \hat{\mathbf{d}}_r^{(k)} \hat{\mathbf{c}}_r^{(k+1)\top} \\ \mathbf{X} &= \mathbf{X}_i + \hat{\mathbf{d}}_i^{(k)} \hat{\mathbf{c}}_i^{(k)\top} + \mathbf{E}_i^{(k+1)}\end{aligned}\quad (3.4.79)$$

则 \mathbf{c}_i 的第 $k+1$ 次迭代值经恒模映射后的值为

$$\begin{aligned}\tilde{\mathbf{c}}_i^{(k+1)} &= \arg \min_{\mathbf{c}_i^{(k)}} \|\mathbf{X} - \mathbf{X}_i - \hat{\mathbf{d}}_i^{(k)} \hat{\mathbf{c}}_i^{(k)\top}\|_{\text{F}}^2 \\ &= \left(\frac{(\hat{\mathbf{d}}_i^{(k)})^{\text{H}}}{\|\hat{\mathbf{d}}_i^{(k)}\|_2^2} (\mathbf{X} - \mathbf{X}_i) \right)^{\text{T}}\end{aligned}\quad (3.4.80)$$

$$\hat{\mathbf{c}}_i^{(k+1)} = \text{proj}_{\text{CM}}(\tilde{\mathbf{c}}_i^{(k+1)}) \quad (3.4.81)$$

将 $\hat{\mathbf{C}}^{(k)}$ 的每一列求解出后，立刻更新至 $\hat{\mathbf{C}}^{(k)}$ 中，当 $i=R$ 时， $\hat{\mathbf{C}}^{(k)}$ 的所有列更新完毕，此时，得到 $\hat{\mathbf{C}}^{(k+1)}$ ， $\hat{\mathbf{C}}^{(k+1)}$ 是一个恒模矩阵。 \mathbf{C} 矩阵按列进行更新，其过程与连续干扰抵消（SIC）过程类似。将 $\mathbf{C}^{(k+1)}$ 带入式(3.4.69)和式(3.4.70)中更新矩阵 \mathbf{A} 和矩阵 \mathbf{B} 。下面给出 TALSSIC 算法的实现。

步骤 1：初始化承载矩阵 $\hat{\mathbf{A}}^{(0)}$ 、 $\hat{\mathbf{B}}^{(0)}$ 、 $\hat{\mathbf{C}}^{(0)}$ ， $k=0$ 。

步骤 2： $k=k+1$ 。

步骤 3：利用式(3.4.71)～式(3.4.81)及 \mathbf{X} 、 $\hat{\mathbf{A}}^{(k-1)}$ 、 $\hat{\mathbf{B}}^{(k-1)}$ ，计算 \mathbf{C} 的第 k 次迭代值 $\hat{\mathbf{C}}^{(k)}$ 。

步骤 4：利用式(3.4.69)及 \mathbf{Y} 、 $\hat{\mathbf{B}}^{(k)}$ 、 $\hat{\mathbf{C}}^{(k-1)}$ ，计算 \mathbf{A} 的第 k 次迭代值 $\hat{\mathbf{A}}^{(k)}$ 。

步骤 5：利用式(3.4.70)及 \mathbf{Z} 、 $\hat{\mathbf{C}}^{(k)}$ 、 $\hat{\mathbf{A}}^{(k)}$ ，计算 \mathbf{B} 的第 k 次迭代值 $\hat{\mathbf{B}}^{(k)}$ 。

步骤 6：重复步骤 2～5，令

$$\varepsilon = \|\mathbf{X} - (\hat{\mathbf{A}}^{(k-1)} \odot \hat{\mathbf{B}}^{(k-1)}) \hat{\mathbf{C}}^{(k-1)\top}\|_{\text{F}}^2 - \|\mathbf{X} - (\hat{\mathbf{A}}^{(k)} \odot \hat{\mathbf{B}}^{(k)}) \hat{\mathbf{C}}^{(k)\top}\|_{\text{F}}^2$$

当 ε 小于一个很小的数时，迭代结束。

TALSSIC 算法对文献[14]的思想进行了扩展。下面，说明 TALSSIC 算法的单调收敛性。由于承载矩阵 \mathbf{A} 和 \mathbf{B} 的求解均为标准的最小二乘过程，它们均可以使得代价函数式(3.4.66)下降或保持，因此，要证明 TALSSIC 算法单调收敛，仅须说明 \mathbf{C} 的求解过程的单调收敛性。首先，考虑下面的约束最小二乘过程：

$$\hat{\mathbf{c}} = \arg \min_{\mathbf{c} \in \Omega} \|\mathbf{X} - \mathbf{d}\mathbf{c}^{\text{T}}\|_{\text{F}}^2 \quad (3.4.82)$$

式(3.4.82)给出的约束最小二乘过程与下列过程等价：

$$\bar{\mathbf{c}} = \arg \min \|\mathbf{X} - \mathbf{d}\mathbf{c}^{\text{T}}\|_{\text{F}}^2 \quad (3.4.83)$$

$$\hat{\mathbf{c}} = \arg \min_{\mathbf{c} \in \Omega} \|\bar{\mathbf{c}} - \mathbf{c}\|_{\text{F}}^2 \quad (3.4.84)$$

式 (3.4.84) 中, $\bar{\mathbf{c}}$ 是式 (3.4.83) 的非约束最小二乘解, 根据式 (3.4.84) 求 \mathbf{c} 的约束最小二乘解等价于式 (3.4.82) 的约束最小二乘解。因此, 通过式 (3.4.80) 和式 (3.4.81) 得到的 $\hat{\mathbf{c}}_i^{(k+1)}$ 是式 (3.4.79) 的恒模约束最小二乘解, 且下列不等式成立:

$$\left\| \mathbf{X} - \mathbf{X}_i - \hat{\mathbf{d}}_i^{(k)} \hat{\mathbf{c}}_i^{(k+1)\text{T}} \right\|_{\text{F}}^2 = \min_{\|\mathbf{c}_i\|=1} \left\| \mathbf{X} - \mathbf{X}_i - \hat{\mathbf{d}}_i^{(k)} \hat{\mathbf{c}}_i^{\text{T}} \right\|_{\text{F}}^2 \leq \left\| \mathbf{X} - \mathbf{X}_i - \hat{\mathbf{d}}_i^{(k)} \hat{\mathbf{c}}_i^{(k)\text{T}} \right\|_{\text{F}}^2 \quad (3.4.85)$$

于是, 有

$$\begin{aligned} f(\mathbf{D}^{(k+1)}, \mathbf{C}^{(k+1)}; \mathbf{X}) &= \left\| \mathbf{X} - \mathbf{D}^{(k+1)} \mathbf{C}^{(k+1)\text{T}} \right\|_{\text{F}}^2 \\ &= \left\| \mathbf{X} - \sum_{r=1}^R \hat{\mathbf{d}}_r^{(k+1)} \hat{\mathbf{c}}_r^{(k+1)\text{T}} \right\|_{\text{F}}^2 \\ &= \left\| \mathbf{X} - \mathbf{X}_{R-1} - \hat{\mathbf{d}}_R^{(k+1)} \hat{\mathbf{c}}_R^{(k+1)\text{T}} \right\|_{\text{F}}^2 \\ &= \min \left\| \mathbf{X} - \mathbf{X}_{R-1} - \hat{\mathbf{d}}_R^{(k+1)} \hat{\mathbf{c}}^{\text{T}} \right\|_{\text{F}}^2 \\ &\leq \left\| \mathbf{X} - \mathbf{X}_{R-1} - \hat{\mathbf{d}}_R^{(k+1)} \hat{\mathbf{c}}_R^{(k)\text{T}} \right\|_{\text{F}}^2 \\ &= \left\| \mathbf{X} - \mathbf{X}_{R-2} - \hat{\mathbf{d}}_{R-1}^{(k+1)} \hat{\mathbf{c}}_{R-1}^{(k+1)\text{T}} \right\|_{\text{F}}^2 \\ &= \min \left\| \mathbf{X} - \mathbf{X}_{R-2} - \hat{\mathbf{d}}_{R-1}^{(k+1)} \hat{\mathbf{c}}^{\text{T}} \right\|_{\text{F}}^2 \\ &\leq \left\| \mathbf{X} - \mathbf{X}_{R-2} - \hat{\mathbf{d}}_{R-1}^{(k+1)} \hat{\mathbf{c}}_{R-1}^{(k)\text{T}} \right\|_{\text{F}}^2 \\ &\quad \vdots \\ &= \left\| \mathbf{X} - \mathbf{X}_1 - \hat{\mathbf{d}}_1^{(k+1)} \hat{\mathbf{c}}_1^{(k+1)\text{T}} \right\|_{\text{F}}^2 \\ &\leq \left\| \mathbf{X} - \mathbf{X}_1 - \hat{\mathbf{d}}_1^{(k+1)} \hat{\mathbf{c}}_1^{(k)\text{T}} \right\|_{\text{F}}^2 \\ &= \left\| \mathbf{X} - \sum_{r=1}^R \hat{\mathbf{d}}_r^{(k)} \hat{\mathbf{c}}_r^{(k)\text{T}} \right\|_{\text{F}}^2 \\ &= \left\| \mathbf{X} - \mathbf{D}^{(k)} \mathbf{C}^{(k)\text{T}} \right\|_{\text{F}}^2 \\ &= f(\mathbf{D}^{(k)}, \mathbf{C}^{(k)}; \mathbf{X}) \end{aligned} \quad (3.4.86)$$

上述推导过程表明, 第 $k+1$ 次迭代得到的代价函数值始终小于或等于第 k 次迭代得到的代价函数值, \mathbf{C} 矩阵的更新过程仅会使代价函数下降或保持, 因此, TALSSIC 算法是单调收敛的。

3.5 PARAFAC 分解的 CRB 分析

本节将对 3.4 节给出的 PARAFAC 模型的信号处理算法的理论性能进行分析。经结构约束后参数估计算法的理论性能可通过求解 CRB 得出。John D. 最早在文献[15]中给出了参数在等式约束条件下 CCRB 的求解方法, Petre Stoica 对此过程进行了发展, 给出了更为简单的 CRB 的表达式^[16]。本节对 3.3 节给出的结构约束三线性分解的 CRB 进行推导和分析, 给出了不同的约束条件下三线性分解的 CRB, 以此说明约束条件在三线性分解中的作用。

3.5.1 三线性分解的 CRB 求解

基于 PARAFAC 分解的信号处理算法的理论性能可以通过三线性分解的 CRB 进行分析^[17], 在 CRB 的推导过程中, 需要使用 PARAFAC 模型的三个剖面形式。含有噪声的 PARAFAC 模型的三个剖面形式如式 (3.5.1) ~ 式 (3.5.3) 所示^[17]:

$$\mathbf{X} = (\mathbf{A} \odot \mathbf{B}) \mathbf{C}^T + \mathbf{N}_3 \quad (3.5.1)$$

$$\mathbf{Y} = (\mathbf{B} \odot \mathbf{C}) \mathbf{A}^T + \mathbf{N}_1 \quad (3.5.2)$$

$$\mathbf{Z} = (\mathbf{C} \odot \mathbf{A}) \mathbf{B}^T + \mathbf{N}_2 \quad (3.5.3)$$

其中, $\mathbf{A} \in \mathbb{C}^{I \times R}$ 、 $\mathbf{B} \in \mathbb{C}^{J \times R}$ 、 $\mathbf{C} \in \mathbb{C}^{K \times R}$ 是 PARAFAC 模型的三个承载矩阵。在下面的推导过程中, 称 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 为参数矩阵, 其元素为需要估计的复值参数。文献[17]将复参数表示为未知参数与其共轭相组合的形式, 即

$$\boldsymbol{\theta} = [\mathbf{a}_1^T, \dots, \mathbf{a}_I^T, \mathbf{b}_1^T, \dots, \mathbf{b}_J^T, \mathbf{c}_1^T, \dots, \mathbf{c}_K^T, \mathbf{c}_1^H, \dots, \mathbf{c}_K^H]^T \quad (3.5.4)$$

其中, $\mathbf{a}_n, \mathbf{b}_n, \mathbf{c}_n$ 分别为 $\mathbf{A}^T, \mathbf{B}^T, \mathbf{C}^T$ 的第 n 列, 在此基础上推导了三线性分解的 CRB 表达式。本节利用文献[18]中的方法对未知参数 $\boldsymbol{\theta}$ 进行设定, 将复参数的实部与虚部分开考虑, 即

$$\boldsymbol{\theta} = [\bar{\mathbf{a}}_1^T, \tilde{\mathbf{a}}_1^T, \dots, \bar{\mathbf{a}}_I^T, \tilde{\mathbf{a}}_I^T, \bar{\mathbf{b}}_1^T, \tilde{\mathbf{b}}_1^T, \dots, \bar{\mathbf{b}}_J^T, \tilde{\mathbf{b}}_J^T, \bar{\mathbf{c}}_1^T, \tilde{\mathbf{c}}_1^T, \dots, \bar{\mathbf{c}}_K^T, \tilde{\mathbf{c}}_K^T]^T \quad (3.5.5)$$

其中, $\bar{\mathbf{a}}$ 表示 \mathbf{a} 的实部, 即 $\bar{\mathbf{a}} = \text{Re}(\mathbf{a})$, $\tilde{\mathbf{a}}$ 表示 \mathbf{a} 的虚部, 即 $\tilde{\mathbf{a}} = \text{Im}(\mathbf{a})$, 在下面的讨论中, $(\bar{\bullet})$ 表示矩阵或向量的实数部分, $(\tilde{\bullet})$ 表示矩阵或向量的虚数部分。此时, 参数 $\boldsymbol{\theta}$ 为实数向量, 这种表示方法可以显式地表示出参数矩阵 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 中复值元素的实部和虚部, 并对其独立进行操作, 避免了复数运算, 这将在下面的 CRB 的推导中给予便利。

根据 PARAFAC 三个剖面 $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ 的表达式, 三维数据集 \mathbf{X} 的对数似然函数可以表示为下列三种等效的形式^[17]:

$$\begin{aligned} L(\boldsymbol{\theta}) &= -IJK \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{k=1}^K \|\mathbf{x}_k - (\mathbf{A} \odot \mathbf{B}) \mathbf{c}_k\|_2^2 \\ &= -IJK \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{i=1}^I \|\mathbf{y}_i - (\mathbf{B} \odot \mathbf{C}) \mathbf{a}_i\|_2^2 \\ &= -IJK \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{j=1}^J \|\mathbf{z}_j - (\mathbf{C} \odot \mathbf{A}) \mathbf{b}_j\|_2^2 \end{aligned} \quad (3.5.6)$$

其中, $\mathbf{x}_k, \mathbf{y}_i, \mathbf{z}_j$ 分别为 $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ 的第 k, i, j 列。令 $\mathbf{H}_1 = \mathbf{B} \odot \mathbf{C}$, $\mathbf{H}_2 = \mathbf{C} \odot \mathbf{A}$, $\mathbf{H}_3 = \mathbf{A} \odot \mathbf{B}$, 根据文献[18]中的附录 E, $L(\boldsymbol{\theta})$ 对于参数 $\boldsymbol{\theta}$ 的梯度 $\partial L(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$ 为实数, 因此 $\boldsymbol{\theta}$ 的 Fisher 矩阵可以表示为

$$\mathbf{I}(\boldsymbol{\theta}) = E \left[\frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] \left[\frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T = \begin{bmatrix} \boldsymbol{\Psi}_{aa} \boldsymbol{\Psi}_{ab} \boldsymbol{\Psi}_{ac} \\ \boldsymbol{\Psi}_{ab}^T \boldsymbol{\Psi}_{bb} \boldsymbol{\Psi}_{bc} \\ \boldsymbol{\Psi}_{ac}^T \boldsymbol{\Psi}_{bc}^T \boldsymbol{\Psi}_{cc} \end{bmatrix} \quad (3.5.7)$$

下面分块求取 $\mathbf{I}(\boldsymbol{\theta})$, 首先, 给出似然函数 $L(\boldsymbol{\theta})$ 对不同参数向量的梯度表达式^[17]:

$$\begin{aligned}
 \frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{a}}_i} &= \frac{2}{\sigma^2} \operatorname{Re}[\mathbf{H}_1^H \mathbf{N}_1(i)], & i=1, \dots, I \\
 \frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{a}}_i} &= \frac{2}{\sigma^2} \operatorname{Im}[\mathbf{H}_1^H \mathbf{N}_1(i)], & i=1, \dots, I \\
 \frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{b}}_j} &= \frac{2}{\sigma^2} \operatorname{Re}[\mathbf{H}_2^H \mathbf{N}_2(j)], & j=1, \dots, J \\
 \frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{b}}_j} &= \frac{2}{\sigma^2} \operatorname{Im}[\mathbf{H}_2^H \mathbf{N}_2(j)], & j=1, \dots, J \\
 \frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{c}}_k} &= \frac{2}{\sigma^2} \operatorname{Re}[\mathbf{H}_3^H \mathbf{N}_3(k)], & k=1, \dots, K \\
 \frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{c}}_k} &= \frac{2}{\sigma^2} \operatorname{Im}[\mathbf{H}_3^H \mathbf{N}_3(k)], & k=1, \dots, K
 \end{aligned} \tag{3.5.8}$$

其中, $\mathbf{N}_1(i), \mathbf{N}_2(j), \mathbf{N}_3(k)$ 分别是噪声切面矩阵 \mathbf{N}_1 、 \mathbf{N}_2 、 \mathbf{N}_3 的第 i, j, k 列。根据式 (3.5.8), 给出似然函数 $L(\boldsymbol{\theta})$ 对不同参数向量的二阶导数表达式:

$$\begin{aligned}
 \mathbf{E} \left[\left[\frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{a}}_{i_1}} \right] \left[\frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{a}}_{i_2}} \right]^T \right] &= \frac{4}{\sigma^4} \frac{1}{2} \operatorname{Re} \left[\mathbf{H}_1^H \mathbf{E} \left[\mathbf{N}_1(i_1) \mathbf{N}_1^H(i_2) \right] \mathbf{H}_1 \right] \\
 \mathbf{E} \left[\left[\frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{a}}_{i_1}} \right] \left[\frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{a}}_{i_2}} \right]^T \right] &= \frac{4}{\sigma^4} \frac{1}{2} \operatorname{Im} \left[\mathbf{H}_1^H \mathbf{E} \left[\mathbf{N}_1(i_1) \mathbf{N}_1^H(i_2) \right] \mathbf{H}_1 \right] \\
 \mathbf{E} \left[\left[\frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{a}}_{i_1}} \right] \left[\frac{\partial L(\boldsymbol{\theta})}{\partial \bar{\mathbf{a}}_{i_2}} \right]^T \right] &= \frac{4}{\sigma^4} \frac{1}{2} \operatorname{Im} \left[\mathbf{H}_1^H \mathbf{E} \left[\mathbf{N}_1(i_1) \mathbf{N}_1^H(i_2) \right] \mathbf{H}_1 \right] \\
 \mathbf{E} \left[\left[\frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{a}}_{i_1}} \right] \left[\frac{\partial L(\boldsymbol{\theta})}{\partial \tilde{\mathbf{a}}_{i_2}} \right]^T \right] &= \frac{4}{\sigma^4} \frac{1}{2} \operatorname{Re} \left[\mathbf{H}_1^H \mathbf{E} \left[\mathbf{N}_1(i_1) \mathbf{N}_1^H(i_2) \right] \mathbf{H}_1 \right]
 \end{aligned} \tag{3.5.9}$$

令

$$\mathbf{R}_{aa}(i_1, i_2) = \frac{2}{\sigma^4} \mathbf{H}_1^H \mathbf{E} \left[\mathbf{N}_1(i_1) \mathbf{N}_1^H(i_2) \right] \mathbf{H}_1 \tag{3.5.10}$$

其中, 由于不同数据的噪声之间是相互独立的, 因此

$$\mathbf{E} \left[\mathbf{N}_1(i_1) \mathbf{N}_1^H(i_2) \right] = \sigma^2 \mathbf{I}_K \delta_{i_1, i_2} \tag{3.5.11}$$

则 $\boldsymbol{\Psi}_{aa}$ 的元素 $\boldsymbol{\Psi}_{aa}(i_1, i_2)$ 可以表示为

$$\boldsymbol{\Psi}_{aa}(i_1, i_2) = \begin{bmatrix} \bar{\mathbf{R}}_{aa}(i_1, i_2) & -\tilde{\mathbf{R}}_{aa}(i_1, i_2) \\ \tilde{\mathbf{R}}_{aa}(i_1, i_2) & \bar{\mathbf{R}}_{aa}(i_1, i_2) \end{bmatrix}, \quad i_1 = 1, \dots, I, \quad i_2 = 1, \dots, I \tag{3.5.12}$$

$\boldsymbol{\Psi}_{aa}$ 具有下列形式:

$$\boldsymbol{\Psi}_{aa} = \begin{bmatrix} \boldsymbol{\Psi}_{aa}(1,1) & \dots & \boldsymbol{\Psi}_{aa}(1,I) \\ \vdots & \ddots & \vdots \\ \boldsymbol{\Psi}_{aa}(I,1) & \dots & \boldsymbol{\Psi}_{aa}(I,I) \end{bmatrix} \tag{3.5.13}$$

同理, 令

$$\begin{aligned}
 \mathbf{R}_{bb}(j_1, j_2) &= \frac{2}{\sigma^4} \mathbf{H}_2^H \mathbf{E} \left[\mathbf{N}_2(j_1) \mathbf{N}_2^H(j_2) \right] \mathbf{H}_2 \\
 &= \frac{2}{\sigma^4} \mathbf{H}_2^H \sigma^2 \mathbf{I}_I \delta_{j_1, j_2} \mathbf{H}_2 \\
 \mathbf{R}_{cc}(k_1, k_2) &= \frac{2}{\sigma^4} \mathbf{H}_3^H \mathbf{E} \left[\mathbf{N}_3(k_1) \mathbf{N}_3^H(k_2) \right] \mathbf{H}_3 \\
 &= \frac{2}{\sigma^4} \mathbf{H}_3^H \sigma^2 \mathbf{I}_J \delta_{k_1, k_2} \mathbf{H}_3
 \end{aligned} \tag{3.5.14}$$

则 Ψ_{bb}, Ψ_{cc} 的元素 $\Psi_{bb}(j_1, j_2), \Psi_{cc}(k_1, k_2)$ 可以表示为

$$\begin{aligned}
 \Psi_{bb}(j_1, j_2) &= \begin{bmatrix} \bar{\mathbf{R}}_{bb}(j_1, j_2) & -\tilde{\mathbf{R}}_{bb}(j_1, j_2) \\ \tilde{\mathbf{R}}_{bb}(j_1, j_2) & \bar{\mathbf{R}}_{bb}(j_1, j_2) \end{bmatrix}, \quad j_1 = 1, \dots, J, \quad j_2 = 1, \dots, J \\
 \Psi_{cc}(k_1, k_2) &= \begin{bmatrix} \bar{\mathbf{R}}_{cc}(k_1, k_2) & -\tilde{\mathbf{R}}_{cc}(k_1, k_2) \\ \tilde{\mathbf{R}}_{cc}(k_1, k_2) & \bar{\mathbf{R}}_{cc}(k_1, k_2) \end{bmatrix}, \quad k_1 = 1, \dots, K, \quad k_2 = 1, \dots, K
 \end{aligned} \tag{3.5.15}$$

Ψ_{bb}, Ψ_{cc} 具有下列形式:

$$\begin{aligned}
 \Psi_{bb} &= \begin{bmatrix} \Psi_{bb}(1,1) \cdots \Psi_{bb}(1,J) \\ \vdots \quad \ddots \\ \Psi_{bb}(J,1) \cdots \Psi_{bb}(J,J) \end{bmatrix} \\
 \Psi_{cc} &= \begin{bmatrix} \Psi_{cc}(1,1) \cdots \Psi_{cc}(1,K) \\ \vdots \quad \ddots \\ \Psi_{cc}(K,1) \cdots \Psi_{cc}(K,K) \end{bmatrix}
 \end{aligned} \tag{3.5.16}$$

下面给出 $\Psi_{ab}, \Psi_{ac}, \Psi_{bc}$ 的表达式。不同剖面的噪声之间的互相关函数为

$$\begin{aligned}
 \mathbf{U}_{ij} &= \mathbf{E} \left[\mathbf{N}_1(i) \mathbf{N}_2^H(j) \right] \in \mathbf{C}^{JK \times KI}, \quad i = 1, \dots, I \quad j = 1, \dots, J \\
 \mathbf{V}_{ik} &= \mathbf{E} \left[\mathbf{N}_1(i) \mathbf{N}_3^H(k) \right] \in \mathbf{C}^{JK \times KI}, \quad i = 1, \dots, I \quad k = 1, \dots, K \\
 \mathbf{W}_{jk} &= \mathbf{E} \left[\mathbf{N}_2(j) \mathbf{N}_3^H(k) \right] \in \mathbf{C}^{KI \times IJ}, \quad i = 1, \dots, J \quad k = 1, \dots, K
 \end{aligned} \tag{3.5.17}$$

其中, 矩阵 $\mathbf{U}_{ij}, \mathbf{V}_{ik}, \mathbf{W}_{jk}$ 中的元素为下列形式:

$$\begin{aligned}
 \mathbf{U}_{ij}(p, q) &= \begin{cases} \sigma^2, & p = (j-1)K + k, q = (k-1)I + i \\ 0, & \text{其他} \end{cases} \quad k = 1, \dots, K \\
 \mathbf{V}_{ik}(p, q) &= \begin{cases} \sigma^2, & p = (j-1)K + k, q = (i-1)j + J \\ 0, & \text{其他} \end{cases} \quad j = 1, \dots, J \\
 \mathbf{W}_{jk}(p, q) &= \begin{cases} \sigma^2, & p = (j-1)K + k, q = (i-1)j + J \\ 0, & \text{其他} \end{cases} \quad i = 1, \dots, I
 \end{aligned} \tag{3.5.18}$$

记

$$\begin{aligned}
 \mathbf{R}_{ab}(i, j) &= \frac{2}{\sigma^4} \mathbf{H}_1^H \mathbf{E} \left[N_1(i) N_1^H(j) \right] \mathbf{H}_2 = \frac{2}{\sigma^4} \mathbf{H}_1^H \mathbf{U}_{ij} \mathbf{H}_2 \\
 \mathbf{R}_{ac}(i, k) &= \frac{2}{\sigma^4} \mathbf{H}_1^H \mathbf{E} \left[N_1(i) N_3^H(k) \right] \mathbf{H}_3 = \frac{2}{\sigma^4} \mathbf{H}_1^H \mathbf{U}_{ik} \mathbf{H}_3 \\
 \mathbf{R}_{bc}(i, j) &= \frac{2}{\sigma^4} \mathbf{H}_2^H \mathbf{E} \left[N_2(i) N_1^H(j) \right] \mathbf{H}_3 = \frac{2}{\sigma^4} \mathbf{H}_2^H \mathbf{U}_{jk} \mathbf{H}_3
 \end{aligned} \tag{3.5.19}$$

得 $\Psi_{ab}, \Psi_{ac}, \Psi_{bc}$ 中的子块表达式为

$$\Psi_{ab}(i, j) = \begin{bmatrix} \bar{\mathbf{R}}_{ab}(i, j) & -\tilde{\mathbf{R}}_{ab}(i, j) \\ \tilde{\mathbf{R}}_{ab}(i, j) & \bar{\mathbf{R}}_{ab}(i, j) \end{bmatrix}, \quad i=1, \dots, I, \quad j=1, \dots, J \tag{3.5.20}$$

$$\Psi_{ac}(i, k) = \begin{bmatrix} \bar{\mathbf{R}}_{ac}(i, k) & -\tilde{\mathbf{R}}_{ac}(i, k) \\ \tilde{\mathbf{R}}_{ac}(i, k) & \bar{\mathbf{R}}_{ac}(i, k) \end{bmatrix}, \quad i=1, \dots, I, \quad k=1, \dots, K \tag{3.5.21}$$

$$\Psi_{bc}(j, k) = \begin{bmatrix} \bar{\mathbf{R}}_{bc}(j, k) & -\tilde{\mathbf{R}}_{bc}(j, k) \\ \tilde{\mathbf{R}}_{bc}(j, k) & \bar{\mathbf{R}}_{bc}(j, k) \end{bmatrix}, \quad j=1, \dots, J, \quad k=1, \dots, K \tag{3.5.22}$$

得到 $\Psi_{ab}, \Psi_{ac}, \Psi_{bc}$ 的表达式:

$$\Psi_{ab} = \begin{bmatrix} \Psi_{ab}(1,1) & \cdots & \Psi_{ab}(1,J) \\ \vdots & \ddots & \vdots \\ \Psi_{ab}(I,1) & \cdots & \Psi_{ab}(I,J) \end{bmatrix} \tag{3.5.23}$$

$$\Psi_{ac} = \begin{bmatrix} \Psi_{ac}(1,1) & \cdots & \Psi_{ac}(1,K) \\ \vdots & \ddots & \vdots \\ \Psi_{ac}(I,1) & \cdots & \Psi_{ac}(I,K) \end{bmatrix} \tag{3.5.24}$$

$$\Psi_{bc} = \begin{bmatrix} \Psi_{bc}(1,1) & \cdots & \Psi_{bc}(1,K) \\ \vdots & \ddots & \vdots \\ \Psi_{bc}(J,1) & \cdots & \Psi_{bc}(J,K) \end{bmatrix} \tag{3.5.25}$$

根据互相关矩阵的性质, 有

$$\Psi_{ba} = \Psi_{ab}^T, \quad \Psi_{ca} = \Psi_{ac}^T, \quad \Psi_{cb} = \Psi_{bc}^T \tag{3.5.26}$$

将式 (3.5.23) ~ 式 (3.5.26) 代入式 (3.5.8), 得到基于 PARAFAC 模型的三线性分解的 FIM (Fisher information matrix):

$$\mathbf{I}(\theta) = \begin{bmatrix} \Psi_{aa} \Psi_{ab} \Psi_{ac} \\ \Psi_{ab}^T \Psi_{bb} \Psi_{bc} \\ \Psi_{ac}^T \Psi_{bc}^T \Psi_{cc} \end{bmatrix} \tag{3.5.27}$$

在 $\mathbf{I}(\theta)$ 可逆的情况下, 三线性分解的 CRB 矩阵为

$$\mathbf{CRB}(\theta) = \mathbf{I}^{-1}(\theta) = \begin{bmatrix} \mathbf{CRB}_{aa} & \mathbf{CRB}_{ab} & \mathbf{CRB}_{ac} \\ \mathbf{CRB}_{ba} & \mathbf{CRB}_{bb} & \mathbf{CRB}_{bc} \\ \mathbf{CRB}_{ca} & \mathbf{CRB}_{cb} & \mathbf{CRB}_{cc} \end{bmatrix} \tag{3.5.28}$$

3.5.2 约束 CRB 的求解算法

使用文献[17-18]中的思想求解结构约束三线性分解的 CRB。设未知参数矢量 $\theta \in C^{1 \times R}$ ，无约束时 θ 的 Fisher 矩阵为 $J(\theta)$ 。对 θ 施加 K 个等式约束，即

$$f_i(\theta) = 0, \quad i = 1, \dots, K \quad (3.5.29)$$

其中， $K < R$ 。记 $f(\theta) = [f_1(\theta), \dots, f_K(\theta)]$ ，则 $f(\theta)$ 对 θ 的梯度矩阵为

$$F(\theta) = \frac{\partial f(\theta)}{\partial(\theta)} \in C^{K \times R} \quad (3.5.30)$$

其中， $F(\theta)$ 中的矩阵元素为

$$[F(\theta)]_{i,j} = \partial f_i(\theta) / \partial[\theta]_j, \quad i = 1, \dots, K \quad j = 1, \dots, R \quad (3.5.31)$$

假设 $F(\theta)$ 行满秩，记 $U \in C^{R \times (R-K)}$ 为 $F(\theta)$ 的零空间的标准正交基，即 U 满足等式：

$$FU = 0, \quad U^H U = I \quad (3.5.32)$$

有下面的定理。

定理 3.5.1^[16-18]：设 $\hat{\theta}$ 是满足式 (3.5.29) 约束条件下的 θ 的无偏估计，当 $U^T J(\theta) U$ 非奇异时，有

$$E[(\hat{\theta} - \theta)(\hat{\theta} - \theta)^T] \geq U(U^T J(\theta) U)^{-1} U^T \quad (3.5.33)$$

当且仅当 $\hat{\theta} - \theta = U(U^T J(\theta) U)^{-1} U^T \Delta$ 时，等式成立。其中 $\Delta = \partial p(x, \theta) / \partial \theta$ ， $p(x, \theta)$ 为似然函数。

因此，CCRB 的求解过程可以分三步进行：

- ① 求解参数 θ 在无约束时的 Fisher 矩阵 $J(\theta)$ ；
- ② 根据式 (3.5.30)，计算等式约束对于 θ 的梯度矩阵 $F(\theta)$ ，求出它的零空间的标准正交基 U ；
- ③ 根据定理 3.5.1，求出 CCRB 矩阵。

3.5.3 “首行已知”约束下三线性分解的 CRB 求解

三线性分解存在固有的列交换模糊和对角尺度模糊，其中对角尺度模糊使得通过三线性分解得到的参数估计值的幅度无法确定，参数无法完全辨识，此时的 $I(\theta)$ 是一个奇异矩阵。3.5.1 节简单地将三个承载矩阵的首行元素设置为 1 以保证 CRB 存在。本节使用约束条件保证参数可辨识，常用的消除尺度模糊的方法是将 A, B, C 的首行元素约束为已知，记为“首行已知”约束。下面给出“首行已知”约束下三线性分解的 CRB。

令 $a_1 = [\bar{a}_{11}, \dots, \bar{a}_{1R}, \tilde{a}_{11}, \dots, \tilde{a}_{1R}]$, $b_1 = [\bar{b}_{11}, \dots, \bar{b}_{1R}, \tilde{b}_{11}, \dots, \tilde{b}_{1R}]$, $c_1 = [\bar{c}_{11}, \dots, \bar{c}_{1R}, \tilde{c}_{11}, \dots, \tilde{c}_{1R}]$ 分别表示矩阵 A, B, C 的第一行元素，将“首行已知”结构用等式约束来表示，即

$$\begin{aligned}
 f_i(\boldsymbol{\theta}) &= \hat{a}_{li} - \bar{a}_{li} = 0 \\
 f_{R+i}(\boldsymbol{\theta}) &= \hat{a}_{li} - \tilde{a}_{li} = 0 \\
 f_{2R+i}(\boldsymbol{\theta}) &= \hat{b}_{li} - \bar{b}_{li} = 0 \\
 f_{3R+i}(\boldsymbol{\theta}) &= \hat{b}_{li} - \tilde{b}_{li} = 0 \\
 f_{4R+i}(\boldsymbol{\theta}) &= \hat{c}_{li} - \bar{c}_{li} = 0 \\
 f_{5R+i}(\boldsymbol{\theta}) &= \hat{c}_{li} - \tilde{c}_{li} = 0 \\
 i &= 1, 2, \dots, R
 \end{aligned} \tag{3.5.34}$$

构造等式约束向量:

$$\mathbf{f}_{\text{FRK}}(\boldsymbol{\theta}) = \begin{bmatrix} f_1(\boldsymbol{\theta}) \\ f_2(\boldsymbol{\theta}) \\ \vdots \\ f_{6R}(\boldsymbol{\theta}) \end{bmatrix} \tag{3.5.35}$$

根据式 (3.5.30), 梯度矩阵 $\mathbf{F}_{\text{FRK}}(\boldsymbol{\theta})$ 为 $\mathbf{f}_{\text{FRK}}(\boldsymbol{\theta})$ 对 $\boldsymbol{\theta}$ 的导数:

$$\mathbf{F}_{\text{FRK}}(\boldsymbol{\theta}) = \frac{\partial \mathbf{f}_{\text{FRK}}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \begin{bmatrix} \mathbf{I}_{2R} & \mathbf{0}_{2R \times 2R(I-1)} & \mathbf{0}_{2R \times 2R(J+K)} \\ \mathbf{0}_{2R \times 2R(I+K)} & \mathbf{I}_{2R} & \mathbf{0}_{2R \times 2R(J-1)} \\ \mathbf{0}_{2R \times 2R(I+J)} & \mathbf{0}_{2R \times 2R(K-1)} & \mathbf{I}_{2R} \end{bmatrix} \tag{3.5.36}$$

$\mathbf{F}_{\text{FRK}}(\boldsymbol{\theta})$ 的零空间的标准正交基 \mathbf{U}_{FRK} 具有以下结构:

$$\mathbf{U}_{\text{FRK}} = \begin{bmatrix} \boldsymbol{\Phi}_{\text{FRK}}^a & \mathbf{0}_{2RI \times 2R(J-1)} & \mathbf{0}_{2RI \times 2R(K-1)} \\ \mathbf{0}_{2RJ \times 2R(I-1)} & \boldsymbol{\Phi}_{\text{FRK}}^b & \mathbf{0}_{2RJ \times 2R(K-1)} \\ \mathbf{0}_{2RK \times 2R(I-1)} & \mathbf{0}_{2RK \times 2R(J-1)} & \boldsymbol{\Phi}_{\text{FRK}}^c \end{bmatrix} \in \mathbf{R}^{2R(I+J+K) \times 2R(I+J+K-3)} \tag{3.5.37}$$

其中:

$$\boldsymbol{\Phi}_{\text{FRK}}^a = \begin{bmatrix} \mathbf{0}_{2R \times 2R(I-1)} \\ \mathbf{I}_{2R(I-1)} \end{bmatrix}, \quad \boldsymbol{\Phi}_{\text{FRK}}^b = \begin{bmatrix} \mathbf{0}_{2R \times 2R(J-1)} \\ \mathbf{I}_{2R(J-1)} \end{bmatrix}, \quad \boldsymbol{\Phi}_{\text{FRK}}^c = \begin{bmatrix} \mathbf{0}_{2R \times 2R(K-1)} \\ \mathbf{I}_{2R(K-1)} \end{bmatrix} \tag{3.5.38}$$

此时, “首行已知” 约束三线性分解的 CRB 矩阵为

$$\text{CRB}_{\text{FRK}} = \mathbf{U}_{\text{FRK}} (\mathbf{U}_{\text{FRK}}^T \mathbf{I}(\boldsymbol{\theta}) \mathbf{U}_{\text{FRK}})^{-1} \mathbf{U}_{\text{FRK}}^T \tag{3.5.39}$$

虽然 $\mathbf{I}(\boldsymbol{\theta})$ 是奇异矩阵, 但 $\mathbf{U}_{\text{FRK}}^T \mathbf{I}(\boldsymbol{\theta}) \mathbf{U}_{\text{FRK}}$ 可逆, 即 “首行已知” 约束后的 CRB 存在, 此时, 参数可辨识。

在一些信号处理算法中, 为了提高参数辨识的质量, 需要在发送数据的前端添加导频序列。假设 \mathbf{C} 矩阵为数据矩阵, 若导频序列的长度为 P , 则 \mathbf{C} 矩阵的前 P 行数据约束为已知, 此时, 含有导频的参数估计算法的 CRB 可以由上述 “首行已知” 结构的 CRB 导出。

3.5.4 恒模约束下三线性分解的 CRB 求解

假设信号矩阵 \mathbf{C} 中的元素为恒模信号, 即

$$\|c_{ij}\|=1, i=2, \dots, K, j=1, \dots, R \quad (3.5.40)$$

矩阵 \mathbf{C} 的恒模约束可以转化为下列 $(K-1) \times M$ 个等式约束:

$$f_{(j-1)R+i}(\boldsymbol{\theta}) = \bar{c}_{ij}^2 + \tilde{c}_{ij}^2 - 1 = 0, i=2, \dots, K, j=1, \dots, R \quad (3.5.41)$$

令

$$\mathbf{f}_{\text{CM}}(\boldsymbol{\theta}) = \begin{bmatrix} f_{R+1}(\boldsymbol{\theta}) \\ \vdots \\ f_{KR}(\boldsymbol{\theta}) \end{bmatrix} \quad (3.5.42)$$

$\mathbf{f}_{\text{CM}}(\boldsymbol{\theta})$ 的梯度矩阵为下列形式:

$$\mathbf{F}_{\text{CM}}(\boldsymbol{\theta}) = \frac{\partial \mathbf{f}_{\text{CM}}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \begin{bmatrix} 0_{R \times 2R(I+K)} & 0_{R \times 2R} \boldsymbol{\Phi}_{\text{CM}}^2 & \vdots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0_{R \times 2R(I+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_{\text{CM}}^K \end{bmatrix} \quad (3.5.43)$$

其中

$$\boldsymbol{\Phi}_{\text{CM}}^k = \begin{bmatrix} 2\bar{c}_{k1} & 2\tilde{c}_{k1} & & \\ 0 & \ddots & 0 & \ddots & 0 \\ & 2\bar{c}_{kR} & 2\tilde{c}_{kR} & & \end{bmatrix}, k=2, \dots, K, k=2, \dots, K \quad (3.5.44)$$

CM 约束本身并不能消除尺度模糊, 可以引入 3.5.3 节给出的“首行已知”约束以消除模糊, 此时 $\mathbf{F}_{\text{CM}}(\boldsymbol{\theta})$ 转化为

$$\begin{aligned} \mathbf{F}'_{\text{CM}}(\boldsymbol{\theta}) &= \begin{bmatrix} \mathbf{F}_{\text{FRK}}(\boldsymbol{\theta}) \\ \mathbf{F}_{\text{CM}}(\boldsymbol{\theta}) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{2R} & 0_{2R \times 2R(I-1)} & 0_{2R \times 2R(J+K)} \\ 0_{2R \times 2RI} & \mathbf{I}_{2R} & 0_{2R \times 2R(J-1)} & 0_{2R \times 2RK} \\ 0_{2R \times 2R(I+J)} & \mathbf{I}_{2R} & 0_{2R \times 2R(K-1)} \\ 0_{R \times 2R(I+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_{\text{CM}}^2 & \\ \vdots & 0 & \ddots & 0 \\ 0_{R \times 2R(I+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_{\text{CM}}^K \end{bmatrix} \end{aligned} \quad (3.5.45)$$

$\mathbf{F}'_{\text{CM}}(\boldsymbol{\theta})$ 零空间的标准正交基 \mathbf{U}_{CM} 具有以下形式:

$$\mathbf{U}_{\text{CM}} = \begin{bmatrix} \boldsymbol{\Phi}_{\text{FRK}}^a & 0_{2RI \times 2R(J-1)} & 0_{2RI \times R(K-1)} \\ 0_{2RJ \times 2R(J-1)} & \boldsymbol{\Phi}_{\text{FRK}}^b & 0_{2RK \times R(K-1)} \\ 0_{2RK \times 2R(I-1)} & 0_{2RK \times 2R(J-1)} & \boldsymbol{\Phi}_{\text{CM}}^c \end{bmatrix} \quad (3.5.46)$$

其中

$$\Phi_{\text{CM}}^c = \begin{bmatrix} & & 0_{2R \times R(K-1)} & & \\ -\tilde{c}_{21} & & \cdots -\tilde{c}_{K1} & & \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ & -\tilde{c}_{2R} & \cdots & & & -\tilde{c}_{KR} \\ \bar{c}_{21} & & \bar{c}_{N1} & & & \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ & \bar{c}_{2R} & \cdots & & \bar{c}_{KR} & \end{bmatrix} \quad (3.5.47)$$

根据定理 3.5.1, 若 $U_{\text{CM}}^T \mathbf{I}(\boldsymbol{\theta}) U_{\text{CM}}$ 可逆, 则 CM 约束下的 CRB 存在, 由式 (3.5.48) 给出:

$$\text{CRB}_{\text{CM}} = U_{\text{CM}} (U_{\text{CM}}^T \mathbf{I}(\boldsymbol{\theta}) U_{\text{CM}})^{-1} U_{\text{CM}}^T \quad (3.5.48)$$

3.5.5 有限字符约束下三线性分解的 CRB 求解

假设数字信号经四相相移键控调制后具有有限字符性质, 此时, 信号矩阵 \mathbf{C} 中的元素 $c_{ij} \in \Omega$, $i=1, \dots, K$, $j=1, \dots, R$, 其中

$$\Omega = \left\{ \frac{\sqrt{2}}{2} + j\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2} + j\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2} - j\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} - j\frac{\sqrt{2}}{2} \right\} \quad (3.5.49)$$

为有限字符集。根据有限字符集的结构, 有下列等式 (不同的有限字符集, 均可以得到类似的等式):

$$\bar{c}_{ij}^2 = \frac{1}{2}, \tilde{c}_{ij}^2 = \frac{1}{2} \quad (3.5.50)$$

根据式 (3.5.50), 可以将矩阵 \mathbf{C} 的 FA 约束转化为 $2R(K-1)$ 个等式约束, 如下所示:

$$\begin{aligned} f_i(\boldsymbol{\theta}) &= \bar{c}_{2i}^2 - 0.5 = 0, \quad i=1, \dots, R \\ f_{R+i}(\boldsymbol{\theta}) &= \tilde{c}_{2i}^2 - 0.5 = 0, \quad i=1, \dots, R \\ &\vdots \\ f_{(2K-4)R+i}(\boldsymbol{\theta}) &= \bar{c}_{2i}^2 - 0.5 = 0, \quad i=1, \dots, R \\ f_{(2K-3)R+i}(\boldsymbol{\theta}) &= \tilde{c}_{2i}^2 - 0.5 = 0, \quad i=1, \dots, R \end{aligned} \quad (3.5.51)$$

$2R(K-1)$ 个等式约束组成的约束向量为

$$f_{\text{FA}}(\boldsymbol{\theta}) = \begin{bmatrix} f_1(\boldsymbol{\theta}) \\ \vdots \\ f_{2R(K-1)}(\boldsymbol{\theta}) \end{bmatrix} \quad (3.5.52)$$

$f_{\text{FA}}(\boldsymbol{\theta})$ 对于 $\boldsymbol{\theta}$ 的梯度矩阵为

$$F_{\text{FA}}(\boldsymbol{\theta}) = \frac{\partial f_{\text{FA}}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \begin{bmatrix} 0_{R \times 2R(I+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_2^{\text{FA}} & \\ 0_{R \times 2R(I+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_3^{\text{FA}} & 0 \\ \vdots & & 0 & \ddots \\ 0_{R \times 2R(I+K)} & 0_{R \times 2R} & & \boldsymbol{\Phi}_K^{\text{FA}} \end{bmatrix} \quad (3.5.53)$$

其中

$$\boldsymbol{\Phi}_i^{\text{FA}} = \begin{bmatrix} 2\bar{c}_{i1} & & & \\ & \ddots & & \\ & & 2\bar{c}_{iR} & 0 \\ 0 & & 2\bar{c}_{i1} & \ddots \\ & & & 2\bar{c}_{iR} \end{bmatrix}, \quad i = 2, \dots, K \quad (3.5.54)$$

为保证参数可辨识性，同样加入“首行已知”约束以消除尺度模糊，此时：

$$F'_{\text{FA}}(\boldsymbol{\theta}) = \begin{bmatrix} F_{\text{FRK}}(\boldsymbol{\theta}) \\ F_{\text{FA}}(\boldsymbol{\theta}) \end{bmatrix} = \begin{bmatrix} I_{2R} & 0_{2R \times 2R(I-1)} & 0_{2R \times 2R(J+K)} \\ 0_{2R \times 2RI} & I_{2R} & 0_{2R \times 2R(J-1)} & 0_{2R \times 2RK} \\ 0_{2R \times 2R(I+J)} & I_{2R} & 0_{2R \times 2R(K-1)} \\ 0_{R \times 2R(1+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_2^{\text{FA}} \\ 0_{R \times 2R(I+K)} & 0_{R \times 2R} & \boldsymbol{\Phi}_3^{\text{FA}} & 0 \\ \vdots & & 0 & \ddots \\ 0_{R \times 2R(1+K)} & 0_{R \times 2R} & & \boldsymbol{\Phi}_K^{\text{FA}} \end{bmatrix} \quad (3.5.55)$$

$F'_{\text{FA}}(\boldsymbol{\theta})$ 零空间的标准正交基 \mathbf{U}_{FA} 为如下形式：

$$\mathbf{U}_{\text{FA}} = \begin{bmatrix} \boldsymbol{\Phi}_{\text{FRK}}^a & 0_{2RI \times 2R(J-1)} \\ 0_{2RJ \times 2R(I-1)} & \boldsymbol{\Phi}_{\text{FRK}}^b \\ 0_{2RK \times 2R(I-1)} & 0_{2RK \times 2R(J-1)} \end{bmatrix} \quad (3.5.56)$$

根据定理 3.5.1，若 $\mathbf{U}_{\text{FA}}^T \mathbf{I}(\boldsymbol{\theta}) \mathbf{U}_{\text{FA}}$ 可逆，则在 FA 约束情况下，CRB 存在，由下式给出：

$$\text{CRB}_{\text{FA}} = \mathbf{U}_{\text{FA}} (\mathbf{U}_{\text{FA}}^T \mathbf{I}(\boldsymbol{\theta}) \mathbf{U}_{\text{FA}})^{-1} \mathbf{U}_{\text{FA}}^T \quad (3.5.57)$$

3.5.6 四线性分解的 CRB 求解

考虑一个 $I \times J \times K \times L$ 的四线性模型 \mathbf{X} ，其元素

$$x_{i,j,k,l} = \sum_{f=1}^F a_{i,f} b_{j,f} g_{k,f} h_{l,f} + n_{i,j,k,l} \quad (3.5.58)$$

其中 $i=1, \dots, I, j=1, \dots, J, k=1, \dots, K, l=1, \dots, L, a_{i,f}, b_{j,f}, g_{k,f}, h_{l,f} \in \mathbb{C}$ ， $n_{i,j,k,l}$ 是零均值、方差

为 σ^2 的复数高斯噪声。定义矩阵 $\mathbf{A}, \mathbf{B}, \mathbf{G}$ 和 \mathbf{H} ，将 $\underline{\mathbf{X}}$ 展开成四个等价矩阵的表示形式：

$$\mathbf{X}_1 = (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \mathbf{A}^T + \mathbf{N}_1 \quad (3.5.59)$$

$$\mathbf{X}_2 = (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \mathbf{B}^T + \mathbf{N}_2 \quad (3.5.60)$$

$$\mathbf{X}_3 = (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \mathbf{G}^T + \mathbf{N}_3 \quad (3.5.61)$$

$$\mathbf{X}_4 = (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{H}^T + \mathbf{N}_4 \quad (3.5.62)$$

相应地，可以将似然函数写成下列四种等效的形式^[17]：

$$\begin{aligned} L(\underline{\mathbf{X}}) &= \frac{1}{(\pi\sigma^2)^{IJKL}} \exp \left\{ -\frac{1}{\sigma^2} \sum_{i=1}^I \|\mathbf{X}_1(i) - (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \bar{\mathbf{a}}_i\|^2 \right\} \\ &= \frac{1}{(\pi\sigma^2)^{IJKL}} \exp \left\{ -\frac{1}{\sigma^2} \sum_{j=1}^J \|\mathbf{X}_2(j) - (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \bar{\mathbf{b}}_j\|^2 \right\} \\ &= \frac{1}{(\pi\sigma^2)^{IJKL}} \exp \left\{ -\frac{1}{\sigma^2} \sum_{k=1}^K \|\mathbf{X}_3(k) - (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \bar{\mathbf{g}}_k\|^2 \right\} \\ &= \frac{1}{(\pi\sigma^2)^{IJKL}} \exp \left\{ -\frac{1}{\sigma^2} \sum_{l=1}^L \|\mathbf{X}_4(l) - (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \bar{\mathbf{h}}_l\|^2 \right\} \end{aligned} \quad (3.5.63)$$

其中 $\bar{\mathbf{a}}_i$ 表示 \mathbf{A}^T 的第 i 列， $\bar{\mathbf{b}}_j, \bar{\mathbf{g}}_k, \bar{\mathbf{h}}_l$ 类似。为了消除排列模糊得到有意义的界限，假设 $\mathbf{A}, \mathbf{B}, \mathbf{G}$ 和 \mathbf{H} 的第一行都是已知的，那么未知的 $1 \times 2(I+J+K+L-4)F$ 复数参数矢量为

$$\boldsymbol{\theta} = [\bar{\mathbf{a}}_2^T, \dots, \bar{\mathbf{a}}_I^T, \bar{\mathbf{b}}_2^T, \dots, \bar{\mathbf{b}}_J^T, \bar{\mathbf{g}}_2^T, \dots, \bar{\mathbf{g}}_K^T, \bar{\mathbf{h}}_2^T, \dots, \bar{\mathbf{h}}_L^T, \bar{\mathbf{a}}_2^H, \dots, \bar{\mathbf{h}}_L^H] \quad (3.5.64)$$

且其对数似然函数 $f(\boldsymbol{\theta}) = \ln L(\underline{\mathbf{X}})$ 为

$$\begin{aligned} f(\boldsymbol{\theta}) &= -IJKL \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{i=1}^I \|\mathbf{X}_1(i) - (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \bar{\mathbf{a}}_i\|^2 \\ &= -IJKL \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{j=1}^J \|\mathbf{X}_2(j) - (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \bar{\mathbf{b}}_j\|^2 \\ &= -IJKL \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{k=1}^K \|\mathbf{X}_3(k) - (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \bar{\mathbf{g}}_k\|^2 \\ &= -IJKL \ln(\pi\sigma^2) - \frac{1}{\sigma^2} \sum_{l=1}^L \|\mathbf{X}_4(l) - (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \bar{\mathbf{h}}_l\|^2 \end{aligned}$$

复数 FIM 为

$$\boldsymbol{\Omega}(\boldsymbol{\theta}) = \mathbf{E} \left\{ \left(\frac{\partial f(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right)^H \left(\frac{\partial f(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right) \right\} \quad (3.5.65)$$

给出 $f(\boldsymbol{\theta})$ 关于未知参数的偏导：

$$\begin{aligned}
\frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f}} &= \frac{1}{\sigma^2} (X_1(i) - (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \bar{a}_i)^H (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \mathbf{e}_f \\
\frac{\partial f(\boldsymbol{\theta})}{\partial b_{j,f}} &= \frac{1}{\sigma^2} (X_2(j) - (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \bar{b}_j)^H (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \mathbf{e}_f \\
\frac{\partial f(\boldsymbol{\theta})}{\partial g_{k,f}} &= \frac{1}{\sigma^2} (X_3(k) - (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \bar{g}_k)^H (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \mathbf{e}_f \\
\frac{\partial f(\boldsymbol{\theta})}{\partial h_{l,f}} &= \frac{1}{\sigma^2} (X_4(l) - (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \bar{h}_l)^H (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{e}_f \\
\frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f}^*} &= \left(\frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f}} \right)^*, \frac{\partial f(\boldsymbol{\theta})}{\partial b_{j,f}^*} = \left(\frac{\partial f(\boldsymbol{\theta})}{\partial b_{j,f}} \right)^*, \frac{\partial f(\boldsymbol{\theta})}{\partial g_{k,f}^*} = \left(\frac{\partial f(\boldsymbol{\theta})}{\partial g_{k,f}} \right)^*, \frac{\partial f(\boldsymbol{\theta})}{\partial h_{l,f}^*} = \left(\frac{\partial f(\boldsymbol{\theta})}{\partial h_{l,f}} \right)^*
\end{aligned} \tag{3.5.66}$$

其中 \mathbf{e}_f 是第 f 个单位坐标向量。

式 (3.5.65) 中的 FIM $\boldsymbol{\Omega}$ 可以写成

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Psi} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Psi}^* \end{bmatrix} \tag{3.5.67}$$

$\boldsymbol{\Omega}$ 的大小为 $2(I+J+K+L-4)F \times 2(I+J+K+L-4)F$, $\boldsymbol{\Psi}$ 中的元素可以通过式 (3.5.68a) ~ 式 (3.5.68j) 被计算出来:

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial a_{i_2,f_2}} \right\} = \frac{1}{\sigma^2} \mathbf{e}_{f_1}^H (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H})^H (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H}) \mathbf{e}_{f_2} \delta_{i_1,i_2} \tag{3.5.68a}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial b_{j_1,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial b_{j_2,f_2}} \right\} = \frac{1}{\sigma^2} \mathbf{e}_{f_1}^H (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A})^H (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \mathbf{e}_{f_2} \delta_{j_1,j_2} \tag{3.5.68b}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial g_{k_1,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial g_{k_2,f_2}} \right\} = \frac{1}{\sigma^2} \mathbf{e}_{f_1}^H (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B})^H (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \mathbf{e}_{f_2} \delta_{k_1,k_2} \tag{3.5.68c}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial h_{l_1,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial h_{l_2,f_2}} \right\} = \frac{1}{\sigma^2} \mathbf{e}_{f_1}^H (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G})^H (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{e}_{f_2} \delta_{l_1,l_2} \tag{3.5.68d}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial b_{j,f_2}} \right\} = \frac{1}{\sigma^4} \mathbf{e}_{f_1}^H (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H})^H E \{ N_1(i) N_2^H(j) \} (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A}) \mathbf{e}_{f_2} \tag{3.5.68e}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial g_{k,f_2}} \right\} = \frac{1}{\sigma^4} \mathbf{e}_{f_1}^H (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H})^H E \{ N_1(i) N_3^H(k) \} (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \mathbf{e}_{f_2} \tag{3.5.68f}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial a_{i,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial h_{l,f_2}} \right\} = \frac{1}{\sigma^4} \mathbf{e}_{f_1}^H (\mathbf{G} \odot \mathbf{B} \odot \mathbf{H})^H E \{ N_1(i) N_4^H(j) \} (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{e}_{f_2} \tag{3.5.68g}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial b_{j,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial g_{k,f_2}} \right\} = \frac{1}{\sigma^4} \mathbf{e}_{f_1}^H (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A})^H E \{ N_2(j) N_3^H(k) \} (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B}) \mathbf{e}_{f_2} \tag{3.5.68h}$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial \mathbf{b}_{j,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial h_{l,f_2}} \right\} = \frac{1}{\sigma^4} \mathbf{e}_{f_1}^H (\mathbf{H} \odot \mathbf{G} \odot \mathbf{A})^H E \{ \mathbf{N}_2(j) \mathbf{N}_4^H(l) \} (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{e}_{f_2} \quad (3.5.68i)$$

$$E \left\{ \frac{\partial f(\boldsymbol{\theta})}{\partial \mathbf{g}_{k,f_1}^*} \frac{\partial f(\boldsymbol{\theta})}{\partial h_{l,f_2}} \right\} = \frac{1}{\sigma^4} \mathbf{e}_{f_1}^H (\mathbf{A} \odot \mathbf{H} \odot \mathbf{B})^H E \{ \mathbf{N}_3(k) \mathbf{N}_4^H(l) \} (\mathbf{B} \odot \mathbf{A} \odot \mathbf{G}) \mathbf{e}_{f_2} \quad (3.5.68j)$$

写出 $\mathbf{N}_1(i)$ 和 $\mathbf{N}_2(j)$ 的协方差矩阵会比较繁琐，但是它的计算可以通过下面简单的方法完成。首先，构造一个辅助的四向阵列 \mathbf{U} 并用整数 1 到 $JKLM$ 进行如下填充：

$$u_{i,j,k,l} = (i-1)JKL + (j-1)KL + (k-1)L + l$$

通过重新整理数据，可以将该四维阵列用四种不同的矩阵形式表示，比如分别和式 (3.5.59)、式 (3.5.60)、式 (3.5.61) 和式 (3.5.62) 一致的矩阵 $\mathbf{U}_1^{(LJK \times I)}$ 、 $\mathbf{U}_2^{(IKL \times J)}$ 、 $\mathbf{U}_3^{(JLI \times K)}$ 和 $\mathbf{U}_4^{(KLJ \times L)}$ 。现在为了获得 $E\{\mathbf{N}_1(i) \mathbf{N}_2^H(j)\}$ ，可以简单地比较 $\mathbf{U}_1(i)$ 和 $\mathbf{U}_2(j)$ ，如果 $\mathbf{U}_1(m,i)$ 和 $\mathbf{U}_2(n,j)$ 包含相同的整数，那么 $[E\{\mathbf{N}_1(i) \mathbf{N}_2^H(j)\}]_{m,n} = \sigma^2$ ，否则 $[E\{\mathbf{N}_1(i) \mathbf{N}_2^H(j)\}]_{m,n} = 0$ 。用这种方式可以简单地获得所需要的协方差矩阵。

式 (3.5.58) 中四线性模型的任意无偏估计量的协方差的 CRB 由式 (3.5.67) 中 $\boldsymbol{\Omega}$ 的逆给出：

$$\text{CRB}(\boldsymbol{\theta}) = \boldsymbol{\Omega}^{-1} = \begin{bmatrix} \boldsymbol{\Psi}^{-1} & \mathbf{0} \\ \mathbf{0} & (\boldsymbol{\Psi}^{-1})^* \end{bmatrix}$$

3.6 自适应 PARAFAC 分解

高阶张量的 PARAFAC 分解是一个强大的多线性代数工具，在多个学科中越来越流行。已有的平行因子算法对计算有需求，并且都应用在批处理模式下，实时应用存在很多缺陷。当获得连续的数据，研究自适应平行因子算法，可以以较低的复杂度跟踪平行因子分解。本节中，考虑一个三阶张量的 PARAFAC 分解的跟踪问题^[19]，其中有一个维度是“时间”。通过增加一个新的时间维度的切面矩阵，在 $t+1$ 时刻得到的张量是通过时刻 t 的张量。为解决该问题，介绍两种不同的自适应算法。

3.6.1 多线性代数基础

定义 3.6.1 (张量形式的 PARAFAC): $\boldsymbol{\chi} \in C^{I \times J \times K}$ 的平行因子分解，就是将 $\boldsymbol{\chi}$ 分解为最小数量的秩 1 张量

$$\boldsymbol{\chi} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \quad (3.6.1)$$

其中 $\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r$ 是承载矩阵 $\mathbf{A} \in C^{I \times R}$ 、 $\mathbf{B} \in C^{J \times R}$ 、 $\mathbf{C} \in C^{K \times R}$ 的第 r 列。

定义 3.6.2 (矩阵形式的 PARAFAC): $\chi \in C^{I \times J \times K}$ 的三种矩阵形式, 根据式 (3.6.1), 和承载矩阵关系如下:

$$\begin{cases} \mathbf{X}^{(1)} = (\mathbf{A} \odot \mathbf{C}) \mathbf{B}^T \\ \mathbf{X}^{(2)} = (\mathbf{B} \odot \mathbf{A}) \mathbf{C}^T \\ \mathbf{X}^{(3)} = (\mathbf{C} \odot \mathbf{B}) \mathbf{A}^T \end{cases}$$

在一般条件下, 平行因子的一个重要特点是其唯一性。 χ 的 PARAFAC 的分解本质上是唯一的, 如果有矩阵 $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$ 也满足该模型, 那么:

$$\mathbf{A} = \tilde{\mathbf{A}} \mathbf{\Pi} \mathbf{A}_1, \quad \mathbf{B} = \tilde{\mathbf{B}} \mathbf{\Pi} \mathbf{A}_2, \quad \mathbf{C} = \tilde{\mathbf{C}} \mathbf{\Pi} \mathbf{A}_3 \quad (3.6.2)$$

其中 $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$ 是任意对角矩阵, 满足 $\mathbf{A}_1 \mathbf{A}_2 \mathbf{A}_3 = \mathbf{I}_R$, $\mathbf{\Pi}$ 是任意排列矩阵。一种特殊情形就是两个承载矩阵是列满秩的, 第三个不包含共线列。在这种情形下, PARAFAC 除了微小的不确定性, 是唯一的。在一个矩阵列满秩, 其余都满秩的情形下, 导出了唯一性的界。结果总结于如下定理。

定理 3.6.1: 相对于 $C^{I \times R + K \times B}$ 中的 Lebesgue 测度, 假设 $\mathbf{A} \in C^{I \times R}$ 、 $\mathbf{C} \in C^{K \times R}$ 是从一个连续共同分布中取出来的, $\mathbf{B} \in C^{J \times R}$ 是列满秩的。如果

$$R \geq 2 \text{ 和 } I(I-1)K(K-1) \geq 2R(R-1) \quad (3.6.3)$$

那么 χ 的 PARAFAC 的分解本质上是唯一的, 几乎可以确定。

注意到若 $R=1$, PARAFAC 分解总是 (平凡) 唯一的。

3.6.2 问题阐述

考虑一个三阶张量 $\chi(t) \in C^{I \times J(t) \times K}$ 的 PARAFAC 分解, 在时刻 t

$$\mathbf{X}^{(1)}(t) \cong \mathbf{H}(t) \mathbf{B}^T(t) \quad (3.6.4)$$

其中, $\mathbf{H}(t) \stackrel{\text{def}}{=} \mathbf{A}(t) \odot \mathbf{C}(t)$, 维数为 $IK \times R$ 。 $\mathbf{B}(t) \in C^{J(t) \times R}$ 的维数随时间增长, $\mathbf{X}^{(1)}(t)$ 是 $\chi(t)$ 的矩阵形式, 维度为 $IK \times J(t)$ 。

实际上, 上述模型可以解释为: $\mathbf{B}(t)$ 中含有 $J(t)$ 个连续的数据矢量 (R 个传输信号的采样), $\mathbf{H}(t)$ 是上述数据矢量的变换矩阵, 并且该矩阵是时变和未知的。 $\mathbf{X}^{(1)}(t)$ 表示观测矢量。

在三阶张量的第二个维度 (即时间维) $\chi(t)$ 附加一个新的观测切片得到新的矩阵 $\chi(t+1) \in C^{I \times J(t+1) \times K}$, 如图 3.6.1 所示, 其中 $J(t+1) = J(t) + 1$ 。 $\chi(t+1)$ 的 PARAFAC 分解的估计值是

$$\mathbf{X}^{(1)}(t+1) \cong \mathbf{H}(t+1) \mathbf{B}^T(t+1) \quad (3.6.5)$$

其中 $\mathbf{H}(t+1) = \mathbf{A}(t+1) \odot \mathbf{C}(t+1)$, 为了估计新的承载矩阵 $\mathbf{A}(t+1)$ 、 $\mathbf{B}(t+1)$ 、 $\mathbf{C}(t+1)$, 可以在 $\chi(t+1)$ 时使用批处理算法, 使 PARAFAC 模型的拟合达到最优。不过, 虽然这种算

法通过旧的 $\mathbf{A}(t)$ 、 $\mathbf{B}(t)$ 、 $\mathbf{C}(t)$ 的估计值来初始化，加快了收敛速度，但在每一采样时刻，需要考虑整个新的 PARAFAC 分解的计算量，对实时应用来说是不可行的。例如，批处理交替最小二乘（Alternating Least Square, ALS）算法的一个迭代需要三次伪逆运算，即使使用较好的值进行初始化，也往往需要很多迭代才能收敛。因此，PARAFAC 分解的自适应跟踪算法的发展是 PARAFAC 走向实时应用的关键一步。总结这些考虑，本节要着重研究的问题归纳如下^[19]。

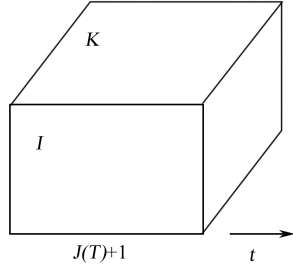


图 3.6.1 在观察的张量上得到的新切片

问题：已知 $I \times J(t) \times K$ 维张量 $\chi(t)$ 的 PARAFAC 分解得到的矩阵 $\mathbf{A}(t)$ 、 $\mathbf{B}(t)$ 、 $\mathbf{C}(t)$ ，因子数为 R ，在 $\chi(t)$ 的第二个维度添加一个切片得到 $I \times J(t+1) \times K$ 维的张量 $\chi(t+1)$ ，通过递归更新求出张量 $\chi(t+1)$ 的 PARAFAC 分解的矩阵 $\mathbf{A}(t+1)$ 、 $\mathbf{B}(t+1)$ 、 $\mathbf{C}(t+1)$ 。

还有以下假设：

- ① 考虑一个未知的慢时变模型，即 \mathbf{A} 和 \mathbf{C} 的所有值在 t 和 $t+1$ 之间发生改变。
- ② 定理 3.6.1 中条件适用于任意 t ，即假设在每一个采样时刻 PARAFAC 分解都具备唯一性。
- ③ R 是已知的或者已经被估计出的，并且不随时间变化。在定理 3.6.1 的条件下， R 可以从 $\mathbf{X}^{(i)}(t)$ 的秩得出，但如果秩随时间变化，即使在矩阵奇异值分解（Singular Value Decomposition, SVD）情形下，跟踪问题也会异常复杂。

3.6.3 基本思想简介

本节在 $\chi(t)$ 给定下给出 $\chi(t+1)$ 的 PARAFAC 分解的自适应计算的第一种方法。由于该方法中包含伪逆和 SVD 操作，复杂度较高，但下面的步骤可以给出即将介绍的自适应算法的框架。

令 $\mathbf{x}(t+1)$ 表示附加到 $\chi(t)$ 的新的切片向量，其维度为 $IK \times 1$ 维，此时有

$$\mathbf{X}^{(i)}(t+1) = [\mathbf{X}^{(i)}(t), \mathbf{x}(t+1)] \quad (3.6.6)$$

考虑 \mathbf{H} 在 t 到 $t+1$ 之间变化平缓，即 $\mathbf{H}(t) \cong \mathbf{H}(t+1)$ ，根据式 (3.6.4)、式 (3.6.5) 和式 (3.6.6) 得到

$$\mathbf{B}^T(t+1) \cong [\mathbf{B}^T(t), \mathbf{b}^T(t+1)] \quad (3.6.7)$$

这表示 \mathbf{B} 近似具有时移结构。使用最小二乘思想给出 $\mathbf{b}(t+1)$ 的初始估计：

$$\mathbf{b}^T(t+1) = \mathbf{H}^+(t) \mathbf{x}(t+1) \quad (3.6.8)$$

根据式 (3.6.7) 中的时移结构，可以构造出 $\mathbf{B}(t+1)$ 。 $\mathbf{H}(t+1)$ 的最小二乘更新为

$$\mathbf{H}(t+1) = \mathbf{X}^{(1)}(t+1)(\mathbf{B}^T(t+1))^+ \quad (3.6.9)$$

给定 $\mathbf{H}(t+1)$ ，就可以代替式 (3.6.8) 中的 $\mathbf{H}(t)$ 来更新 $\mathbf{b}(t+1)$ 。最后，由于 $\mathbf{H}(t+1)$ 是 $\mathbf{A}(t+1) \odot \mathbf{C}(t+1)$ 的估计，此时可以得到 $\mathbf{A}(t+1)$ 和 $\mathbf{C}(t+1)$ 的估计（存在 PARAFAC 分解的固有模糊），实际上， $\mathbf{A}(t+1) \odot \mathbf{C}(t+1)$ 的第 r 列是 $\mathbf{a}_r(t+1) \otimes \mathbf{c}_r(t+1) = \text{vec}(\mathbf{c}_r(t+1) \mathbf{a}_r^T(t+1))$ ，所以它是由一个秩为 1 的矩阵向量化所得到的。因此 $\mathbf{c}_r(t+1)$ 可以通过 $\text{unvec}(\mathbf{a}_r(t+1) \otimes \mathbf{c}_r(t+1))$ 的主左奇异向量乘以相应的奇异值来估计，而 $\mathbf{a}_r(t+1)$ 可以通过主右奇异向量的共轭来估计。重复该过程直到 R 列。

表 3.6.1 给出了算法步骤。要完善充实自适应算法，有以下关键问题需要解决。

表 3.6.1 跟踪 PARAFAC 分解的第一种方法^[19]

步骤 1 假设 $\mathbf{H}(t) = \mathbf{H}(t+1)$ ，得到 $\mathbf{b}(t+1)$ 的第一次估计值： $\mathbf{b}^T(t+1) = \mathbf{H}^+(t) \mathbf{x}(t+1)$
步骤 2 得到 $\mathbf{B}(t+1)$ 的第一次估计值： $\mathbf{B}^T(t+1) = [\mathbf{B}^T(t), \mathbf{b}^T(t+1)]$
步骤 3 估计 $\mathbf{H}(t+1)$ ： $\mathbf{H}(t+1) = \mathbf{X}^{(1)}(t+1)(\mathbf{B}^T(t+1))^+$
步骤 4 由 $\mathbf{H}(t+1)$ 估计 $\mathbf{A}(t+1)$ 和 $\mathbf{C}(t+1)$ For $r=1$ to R , Do $\mathbf{H}_r(t+1) = \text{unvec}([\mathbf{H}(t+1)]_{:,r})$ $[\mathbf{c}_r, \sigma_r, \mathbf{a}_r] = \text{svd}(\mathbf{H}_r(t+1))$ $[\mathbf{C}(t+1)]_{:,r} = \sigma_r \mathbf{c}_r$ and $[\mathbf{A}(t+1)]_{:,r} = \mathbf{a}_r^*$ End
步骤 5 重新估计具有时移结构的 $\mathbf{B}(t+1)$ ： $\begin{cases} \mathbf{b}^T(t+1) = \mathbf{H}^+(t+1) \mathbf{x}(t+1) \\ \mathbf{B}^T(t+1) = [\mathbf{B}^T(t), \mathbf{b}^T(t+1)] \end{cases}$

- ① 观测矩阵 $\mathbf{X}^{(1)}(t)$ 必须被妥善加窗，对过去的观测值进行加权；
- ② 通过递归更新伪逆矩阵；
- ③ 用 SVD 跟踪算法取代传统的 SVD；
- ④ 避免操作的复杂度随着时间增加而增加。

在下一节，着重考虑问题 1，同时说明如何使用指数窗或滑动窗。

3.6.4 窗的选择

1. 指数窗

$\chi(t+1)$ 中的 PARAFAC 元素的估计值可以通过最小化指数窗最小二乘准则 (EWLS) 得到

$$\min_{\mathbf{H}(t+1), \mathbf{B}(t+1)} (\phi^{EW}(t+1)) \quad (3.6.10)$$

其中:

$$\phi^{EW}(t+1) = \sum_{\tau=1}^{t+1} \lambda^{t+1-\tau} \|\mathbf{x}(\tau) - \mathbf{H}(t+1)\mathbf{b}^T(\tau)\|^2 \quad (3.6.11)$$

λ 是遗忘因子。该窗表示在任何时间, 所有以前的观测切片都有不同加权。定义加权观测矩阵:

$$\mathbf{X}_{EW}(t+1) \stackrel{\text{def}}{=} \mathbf{X}^{(1)}(t+1)\mathbf{A}(t+1) \quad (3.6.12)$$

$\mathbf{A}(t+1) = \text{diag}([\lambda^{t/2}, \lambda^{t-1/2}, \dots, \lambda^{1/2}, 1])$ 是加权矩阵。指数窗说明 \mathbf{X}_{EW} 的更新规则如下:

$$\mathbf{X}_{EW}(t+1) = [\lambda^{1/2}\mathbf{X}_{EW}(t), \mathbf{x}(t+1)] \quad (3.6.13)$$

2. 截断窗

对于一个长度 $N > R$ 的截断窗, $\mathbf{X}^{(1)}(t+1)$ 的最后 N 列组成的 $IK \times N$ 的矩阵 $\mathbf{X}_{TW}^{(1)}(t+1)$ 表示为

$$\mathbf{X}_{TW}^{(1)}(t+1) = [\mathbf{x}(t+2-N), \dots, \mathbf{x}(t), \mathbf{x}(t+1)] \quad (3.6.14)$$

定义 $\chi_{TW}(t+1) \in C^{I \times N \times K}$ 为 N 个最新的切片构成的张量, $\mathbf{X}_{TW}^{(1)}(t+1)$ 是其矩阵形式。 $\chi_{TW}(t+1)$ 的 PARAFAC 分解估计值可以通过最小化截断窗最小二乘准则 (TWLS) 得到

$$\min_{\mathbf{H}(t+1), \mathbf{B}_{TW}(t+1)} (\phi^{TW}(t+1)) \quad (3.6.15)$$

其中

$$\phi^{TW}(t+1) = \sum_{\tau=1}^N \lambda^{N-\tau} \|\mathbf{x}(u+\tau) - \mathbf{H}(t+1)\mathbf{b}^T(u+\tau)\|^2 \quad (3.6.16)$$

λ 是遗忘因子, $u = t+1-N$, $\mathbf{B}_{TW}(t+1)$ 由 $\mathbf{B}(t+1)$ 的最后 N 行构成。该窗表示在任何时刻, 最近的 N 个切片都有不同加权。 $\lambda = 1$ 的情况对应于长方形的滑动窗口, $\lambda < 1$ 的情形对应于指数衰减的滑动窗口。

定义加权观测矩阵 $\mathbf{X}_{TW}(t+1)$, $\mathbf{X}_{TW}(t+1)$ 可以从 $\mathbf{X}_{TW}^{(1)}(t+1)$ 得到:

$$\mathbf{X}_{TW}(t+1) \stackrel{\text{def}}{=} \mathbf{X}_{TW}^{(1)}(t+1)\mathbf{A} \quad (3.6.17)$$

$$\mathbf{A} = \text{diag}\left(\left[\lambda^{N-1/2}, \lambda^{N-2/2}, \dots, \lambda^{1/2}, 1\right]\right), \quad \mathbf{X}_{TW} \text{ 的更新准则为}$$

$$\left[\lambda^{N/2} \mathbf{x}(t+1-N), \mathbf{X}_{TW}(t+1)\right] = \left[\lambda^{1/2} \mathbf{X}_{TW}(t), \mathbf{x}(t+1)\right] \quad (3.6.18)$$

给定加窗的观测数据，现在应用表 3.6.1 框架来构建 PARAFAC-SDT 以及 PARAFAC-RLST 算法。

3.6.5 PARAFAC-SDT 算法

1. 预备知识

对于一个指数窗，考虑 $\mathbf{X}_{EW}(t)$ 的以下两种分解：

$$\begin{cases} \mathbf{X}_{EW}(t) = (\mathbf{A}(t) \odot \mathbf{C}(t)) \mathbf{B}^T(t) \mathbf{A}(t) \\ \mathbf{X}_{EW}(t) = \mathbf{U}_{EW}(t) \boldsymbol{\Sigma}_{EW}(t) \mathbf{V}_{EW}^H(t) \end{cases} \quad (3.6.19)$$

第一个分解源于式 (3.6.12) 中的 PARAFAC 分解，用 $\mathbf{X}^{(1)}(t)$ 替代 $\mathbf{X}^{(1)}(t+1)$ 即可得到。第二个是 $\mathbf{X}_{EW}(t)$ 的 SVD 分解， $\mathbf{U}_{EW}(t) \in C^{IK \times R}$ 、 $\boldsymbol{\Sigma}_{EW}(t) \in C^{R \times R}$ 、 $\mathbf{V}_{EW}(t) \in C^{J(t) \times R}$ 。

同理，对于一个截断窗，有

$$\begin{cases} \mathbf{X}_{TW}(t) = (\mathbf{A}(t) \odot \mathbf{C}(t)) \mathbf{B}_{TW}^T(t) \mathbf{A}(t) \\ \mathbf{X}_{TW}(t) = \mathbf{U}_{TW}(t) \boldsymbol{\Sigma}_{TW}(t) \mathbf{V}_{TW}^H(t) \end{cases} \quad (3.6.20)$$

$\mathbf{U}_{TW}(t) \in C^{IK \times R}$ 、 $\boldsymbol{\Sigma}_{TW}(t) \in C^{R \times R}$ 、 $\mathbf{V}_{TW}(t) \in C^{N \times R}$ 。

在定理 3.6.1 的条件下，几乎可以确定， $\mathbf{X}_{EW}(t)$ 、 $\mathbf{X}_{TW}(t)$ 的秩为 R 。因此根据式 (3.6.19)，存在一个非奇异矩阵 $\mathbf{W}_{EW}(t) \in C^{R \times R}$ ：

$$\begin{cases} \mathbf{A}(t) \odot \mathbf{C}(t) = \mathbf{E}_{EW}(t) \mathbf{W}_{EW}(t) \\ \mathbf{B}^T(t) \mathbf{A}(t) = \mathbf{W}_{EW}^{-1}(t) \mathbf{V}_{EW}^H(t) \end{cases} \quad (3.6.21)$$

这里 $\mathbf{E}_{EW}(t) \stackrel{\text{def}}{=} \mathbf{U}_{EW}(t) \boldsymbol{\Sigma}_{EW}(t)$ 。同理，从式 (3.6.20)，存在一个非奇异矩阵 $\mathbf{W}_{TW}(t) \in C^{R \times R}$ ，使得

$$\begin{cases} \mathbf{A}(t) \odot \mathbf{C}(t) = \mathbf{E}_{TW}(t) \mathbf{W}_{TW}(t) \\ \mathbf{B}_{TW}^T(t) \mathbf{A}(t) = \mathbf{W}_{TW}^{-1}(t) \mathbf{V}_{TW}^H(t) \end{cases} \quad (3.6.22)$$

$\mathbf{E}_{TW}(t) \stackrel{\text{def}}{=} \mathbf{U}_{TW}(t) \boldsymbol{\Sigma}_{TW}(t)$ 。矩阵 $\mathbf{W}_{EW}(t)$ 通过式 (3.6.21) 将两个等式联系起来，文献[20]通过 R 个 $R \times R$ 矩阵的联合对角化解决该问题，该方法同样适用于 $\mathbf{W}_{TW}(t)$ 以及式 (3.6.22)。PARAFAC-SDT 的目标是以递归的方式，更新 \mathbf{W}_{EW} 、 \mathbf{W}_{TW} 以及它们的逆。

在时刻 $t+1$ ，式 (3.6.21) 变成

$$\begin{cases} \mathbf{A}(t+1) \odot \mathbf{C}(t+1) = \mathbf{E}_{EW}(t+1) \mathbf{W}_{EW}(t+1) \\ \mathbf{B}^T(t+1) \mathbf{A}(t+1) = \mathbf{W}_{EW}^{-1}(t+1) \mathbf{V}_{EW}^H(t+1) \end{cases} \quad (3.6.23)$$

$$\begin{cases} \mathbf{A}(t+1) \odot \mathbf{C}(t+1) = \mathbf{E}_{TW}(t+1) \mathbf{W}_{TW}(t+1) \\ \mathbf{B}_{TW}^T(t+1) \mathbf{A} = \mathbf{W}_{TW}^{-1}(t+1) \mathbf{V}_{TW}^H(t+1) \end{cases} \quad (3.6.24)$$

PARAFAC-SDT 的主要思想是通过应用 $\mathbf{B}(t+1)$ 的时移结构来将式 (3.6.21) ~ 式 (3.6.24) 联系起来。因此, 利用 $\mathbf{B}(t)$ 、 $\mathbf{B}(t+1)$ 之间的共同的数据块, 就会实现递归更新。在给出这些更新规则之前, 第一步是从 $\mathbf{E}_{EW}(t)$ 、 $\mathbf{V}_{EW}(t)$ 中估计 $\mathbf{E}_{EW}(t+1)$ 、 $\mathbf{V}_{EW}(t+1)$, 例如, 可以对加权矩阵 \mathbf{X}_{EW} 的 SVD 分解进行跟踪。该方法同样适用于 \mathbf{X}_{TW} 。

2. PARAFAC-SDT 的步骤

第一步: SVD 跟踪。首先, 左右子空间都要跟踪。同时, 需要计算正交右子空间向量基, 由于 \mathbf{V} 矩阵的性质会在下一步递归更新 \mathbf{W} 时使用。在指数窗情形下, \mathbf{X}_{EW} 和 \mathbf{V}_{EW} 的维数不断增长。在截断窗情况下, \mathbf{E}_{TW} 和 \mathbf{V}_{TW} 有固定的维数。在两种窗情形下, 可以使用单步执行传统的双向迭代技术更新 \mathbf{E} 和 \mathbf{V} 。

然而, 在指数窗下, 由于 \mathbf{X}_{EW} 维数不断增长, 该技术的复杂度不断增加。虽然, 许多文献提出了自适应算法来针对此种指数窗情形, 但绝大多数都只是跟踪了固定维数对应的子空间而已。可以使用 Bi-SVD1 算法^[21]解决此问题。对于截断窗, 复杂度是固定的, 但双向迭代的主要复杂度是 $16RIKN$ (假设 $R \ll \min(IK, N)$), 文献[22,23]中提出了一些低复杂度的针对滑动窗 SVD 跟踪问题的算法。

第二步: 递归更新 \mathbf{W} 和 \mathbf{W}^{-1} 。为了给出 \mathbf{W} 和 \mathbf{W}^{-1} 的递归更新, 需要用到 \mathbf{V} 中的右子空间基向量的正交性, 从第一步可以得到 \mathbf{V} , 对于两种窗, 有下式:

$$\begin{cases} \mathbf{B}^T(t) \mathbf{A}(t) = \mathbf{W}^{-1}(t) \mathbf{V}^H(t) \\ \mathbf{B}^T(t+1) \mathbf{A}(t+1) = \mathbf{W}^{-1}(t+1) \mathbf{V}^H(t+1) \end{cases} \quad (3.6.25)$$

这里下标 TW 和 EW 被忽略, 在 TW 时, $\mathbf{A}(t+1)$ 变为 \mathbf{A} 。在该系统中, 已知 $\mathbf{B}(t)$ 、 $\mathbf{W}^{-1}(t)$ 、 $\mathbf{V}(t)$, 并且从第一步得到 $\mathbf{V}(t+1)$ 。因为时移结构, $\mathbf{B}(t)$ 和 $\mathbf{B}(t+1)$ 有共同的块, 该块用 $\tilde{\mathbf{B}}(t)$ 表示。

如果第一步中用到的是指数窗, $\mathbf{V}_{EW}(t)$ 和 $\mathbf{V}_{EW}(t+1)$ 的维数分别为 $J(t) \times R$ 和 $J(t+1) \times R$, $\mathbf{B}(t)$ 和 $\mathbf{B}(t+1)$ 维数同样分别为 $J(t) \times R$ 和 $J(t+1) \times R$, 在该情形下, 共同块 $\tilde{\mathbf{B}}(t)$ 就是 $\mathbf{B}(t)$ 自身。

对于 $N > R$ 的截断窗, \mathbf{V}_{TW} 和 \mathbf{B}_{TW} 在每个时刻维数都是 $N \times R$ 。该情况下, 共同块 $\tilde{\mathbf{B}}(t)$ 维数 $N-1 \times R$, 并且有

$$\tilde{\mathbf{B}}(t) = [\mathbf{B}_{TW}(t)]_{2:N,:} = [\mathbf{B}_{TW}(t+1)]_{1:N-1,:}$$

根据考虑的窗函数, 定义以下矩阵。

(EW):

$$\begin{cases} \tilde{\mathbf{V}}(t) = \mathbf{V}_{EW}(t) \\ \tilde{\mathbf{V}}(t+1) = [\mathbf{V}_{EW}(t+1)]_{1:J(t),:} \\ \tilde{\mathbf{v}}(t) = [\mathbf{V}_{EW}(t+1)]_{J(t+1),:} \end{cases}$$

(TW):

$$\begin{cases} \tilde{\mathbf{V}}(t) = [\mathbf{V}_{TW}(t)]_{2:N,:} \\ \tilde{\mathbf{v}}(t) = [\mathbf{V}_{TW}(t)]_{1,:} \\ \tilde{\mathbf{V}}(t+1) = [\mathbf{V}_{TW}(t+1)]_{1:N-1,:} \\ \tilde{\mathbf{v}}(t+1) = [\mathbf{V}_{TW}(t+1)]_{N,:} \end{cases}$$

对于上述两种窗，由式 (3.6.25) 中共同块 $\tilde{\mathbf{B}}(t)$ 得到

$$\lambda^{1/2} \mathbf{W}^{-1}(t) \tilde{\mathbf{V}}^H(t) = \mathbf{W}^{-1}(t+1) \tilde{\mathbf{V}}^H(t+1) \quad (3.6.26)$$

由此可得

$$\mathbf{W}^{-1}(t+1) = \lambda^{1/2} \mathbf{W}^{-1}(t) \tilde{\mathbf{V}}^H(t) (\tilde{\mathbf{V}}^H(t+1))^+ \quad (3.6.27)$$

同样有

$$\mathbf{W}(t+1) = \lambda^{-1/2} \tilde{\mathbf{V}}^H(t+1) (\tilde{\mathbf{V}}^H(t))^+ \mathbf{W}(t) \quad (3.6.28)$$

现在的主要工作是要避免式 (3.6.27)、式 (3.6.28) 中的求伪逆操作的计算量。对于两种窗，第一步得到的正交矩阵 $\mathbf{V}(t+1)$ 去掉最后一行 $\tilde{\mathbf{v}}(t+1)$ 后得到 $\tilde{\mathbf{V}}(t+1)$ ，对于秩 1 更新，根据矩阵求逆引理，有

$$(\tilde{\mathbf{V}}^H(t+1))^+ = \tilde{\mathbf{V}}(t+1) \left(\mathbf{I}_R + \frac{\tilde{\mathbf{v}}^H(t+1) \tilde{\mathbf{v}}(t+1)}{1 - \|\tilde{\mathbf{v}}(t+1)\|^2} \right) \quad (3.6.29)$$

代入式 (3.6.27) 求得 \mathbf{W}^{-1} 的递归更新。

定义 \mathbf{W} 的一个递归更新。在指数窗下， $\tilde{\mathbf{V}}(t) = \mathbf{V}_{EW}(t)$ 是酉矩阵，有 $(\tilde{\mathbf{V}}^H(t))^+ = \mathbf{V}_{EW}(t)$ 。在截断窗下， $\tilde{\mathbf{V}}(t)$ 是从正交矩阵 $\mathbf{V}_{TW}(t)$ 去掉第一行 $\tilde{\mathbf{v}}(t)$ 得到的。根据相同的矩阵求逆引理，得到

$$(\tilde{\mathbf{V}}^H(t))^+ = \tilde{\mathbf{V}}(t) \left(\mathbf{I}_R + \frac{\tilde{\mathbf{v}}^H(t) \tilde{\mathbf{v}}(t)}{1 - \|\tilde{\mathbf{v}}(t)\|^2} \right) \quad (3.6.30)$$

代入式 (3.6.28) 求得 \mathbf{W} 的递归更新。

第三步： \mathbf{A} 和 \mathbf{C} 的更新：首先，构造矩阵 $\mathbf{H}(t+1) = \mathbf{E}(t+1) \mathbf{W}(t+1)$ ，根据第一步， $\mathbf{E}(t+1) = \mathbf{U}(t+1) \mathbf{\Sigma}(t+1)$ ，再根据第二步得到 $\mathbf{W}(t+1)$ 。 $\mathbf{H}(t+1)$ 是 $\mathbf{A}(t+1) \odot \mathbf{C}(t+1)$ 的估计，根据表 3.6.1 中第四步的过程，可以求出 $\mathbf{A}(t+1)$ 和 $\mathbf{C}(t+1)$ 。相反，通过跟踪 $\mathbf{H}_r(t+1)$ 的第一个左和右奇异向量来完成 \mathbf{c}_r 和 \mathbf{a}_r 的递归更新。可以通过下面的步骤，使用 Bi-SVD 迭代法求解这些极端奇异向量。

$$\begin{aligned} \mathbf{a}_r^*(t+1) &= \mathbf{H}_r^H(t+1) \mathbf{c}_r(t) \\ \mathbf{c}_r(t+1) &= \mathbf{H}_r(t+1) \mathbf{a}_r(t+1) / \|\mathbf{H}_r(t+1) \times \mathbf{a}_r(t+1)\| \end{aligned}$$

第四步： \mathbf{B} 的更新，更新过的新的行 $\mathbf{b}^T(t+1) = \mathbf{W}^{-1}(t+1) \tilde{\mathbf{v}}^H(t+1)$ ，加到 $\mathbf{B}(t)$ 上来构造 $\mathbf{B}(t+1)$ ，式 (3.6.27) 给出了 $\mathbf{W}^{-1}(t+1)$ 。

3. 复杂度

表 3.6.2 给出了 PARAFAC-SDT 的总的复杂度和每一步的复杂度。复杂度用实际浮点操作数量来计算。例如， d 维复向量的标量积包含 $8d$ 的浮点操作。算法中如果采用指数窗，由于 $\mathbf{V}(t)$ 的横向维度不断增加，步骤 1 中复杂度将取决于时间。从该意义上说，最好是将 PARAFAC-SDT 和截断窗结合起来。在后一种情形下，总体复杂度为 $16R^3 + R^2(31IK + 31N + 40) + R(32IK + 10K + 20)$ 。同样，在最近的 N 个切片构成的张量上，批处理 ALS 算法的 1 次迭代的复杂度为 $88R^3 + R^2(64NK + 64IK + 64IN + 24) + R(24INK + 8NK + 8IK + 8IN)$ ，可见 PARAFAC-SDT 的复杂度远低于 ALS。

表 3.6.2 PARAFAC-SDT 算法的总结^[19]

时刻 t 的输入： 旧估计值： $\mathbf{A}(t), \mathbf{B}(t), \mathbf{C}(t), \mathbf{W}(t), \mathbf{W}^{-1}(t), \mathbf{E}(t)$ 和 $\mathbf{V}(t)$ 观测值： $\mathbf{X}^{(1)}(t), \mathbf{X}^{(1)}(t+1) = [\mathbf{X}^{(1)}(t), \mathbf{x}(t+1)]$ 窗口：选择 λ ，在 $\mathbf{X}^{(1)}(t+1)$ 上应用一个 TW 和一个 EW 构造 $\mathbf{X}(t+1) (= \mathbf{X}_{TW}(t+1) \text{ or } \mathbf{X}_{EW}(t+1))$ 在 $t+1$ 时刻更新		
步骤 1：跟踪 $\mathbf{X}(t+1)$ 的 SVD 首选：对于 TW 或 EW ，进行一步 Bi-SVD 迭代 更好的选择： EW ，选择 Bi-SVD[21] 计算正交右子空间，复杂度随着时间线性上升 步骤 1 的输出： $(EW): \mathbf{U}(t+1) \in \mathbb{C}^{KI \times R}, \sum(t+1) \in \mathbb{C}^{R \times R}, \mathbf{V}(t+1) \in \mathbb{C}^{J(t+1) \times R}$ $(TW): \mathbf{U}(t+1) \in \mathbb{C}^{KI \times R}, \sum(t+1) \in \mathbb{C}^{R \times R}, \mathbf{V}(t+1) \in \mathbb{C}^{N \times R}$		
步骤 2：更新 $\mathbf{W}, \mathbf{W}^{-1}$ 构造矩阵： $(EW): \tilde{\mathbf{V}}(t) = \mathbf{V}(t), \tilde{\mathbf{V}}(t+1) = [\mathbf{V}(t+1)]_{1:J(t)+1}, \mathbf{v}(t+1) = [\mathbf{V}(t+1)]_{J(t+1)+1:}$ $(TW): \tilde{\mathbf{V}}(t) = [\mathbf{V}(t)]_{1:N}, \tilde{\mathbf{v}}(t+1) = [\mathbf{V}(t)]_{1:}, \tilde{\mathbf{V}}(t+1) = [\mathbf{V}(t+1)]_{1:N-1}, \tilde{\mathbf{v}}(t+1) = [\mathbf{V}(t+1)]_{N:}$ 更新		
构造 $\mathbf{Z} = \lambda^{-1/2} \tilde{\mathbf{V}}^H(t+1) \mathbf{V}(t)$ 更新 \mathbf{W} $EW: \mathbf{W}(t+1) = \mathbf{Z} \mathbf{W}(t)$ $TW: \mathbf{W}(t+1) = \mathbf{Z} (\mathbf{I}_R + \frac{\tilde{\mathbf{v}}^H(t) \tilde{\mathbf{v}}(t)}{\mathbf{I} - \ \tilde{\mathbf{v}}(t)\ ^2}) \mathbf{W}(t)$ 更新 \mathbf{W}^{-1} EW 或 $TW:$ $\mathbf{W}^{-1}(t+1) = \mathbf{W}^{-1}(t) \mathbf{Z}^H (\mathbf{I}_R + \frac{\tilde{\mathbf{v}}^H(t+1) \tilde{\mathbf{v}}(t+1)}{\mathbf{I} - \ \tilde{\mathbf{v}}(t+1)\ ^2})$	$8R^2 J(t)$ $8R^3$ $8R^3 + 23R^2 + 10R$	$8R^2(N-1)$ $8R^3 + 23R^2 + 10R$ $8R^3 + 23R^2 + 10R$

续表

步骤 3: 更新 \mathbf{A} 、 \mathbf{C}		
更新		
$\mathbf{H}(t+1) = \mathbf{E}(t+1)\mathbf{W}(t+1)$ For $r = 1$ to R Do $\mathbf{H}_r(t+1) = \text{unvec}([\mathbf{H}(t+1)_{:,r}])$ $\mathbf{a}_r^*(t+1) = \mathbf{H}_r^H(t+1)\mathbf{c}_r(t)$ $\mathbf{c}_r(t+1) = \frac{\mathbf{H}_r(t+1)\mathbf{a}_r(t+1)}{\ \mathbf{H}_r(t+1)\mathbf{a}_r(t+1)\ }$	$8R^2 IK$ $R \times$ $8IK$ $8IK + 10K$	$8R^2 IK$ $R \times$ $8IK$ $8IK + 10K$
步骤 4: 更新 \mathbf{B}		
更新		
$\mathbf{b}^T(t+1) = \mathbf{W}^{-1}(t+1)\tilde{\mathbf{v}}^H(t+1)$ $\mathbf{B}^T(t+1) = [\mathbf{B}^T(t), \mathbf{b}^T(t+1)]$	$8R^2$	$8R^2$
输出:		
矩阵 $\mathbf{A}(t+1)$ 、 $\mathbf{B}(t+1)$ 、 $\mathbf{C}(t+1)$ 表示 $\chi(t+1)$ PARAFAC 分解的估计值		

3.6.6 PARAFAC-RLST 算法

1. 算法原理

本节推导 PARAFAC-RLST 跟踪算法（借助于递归最小二乘跟踪的 PARAFAC）。该算法的原理框架是基于表 3.6.1 的。首先，式（3.6.8）给出了 $\mathbf{b}(t+1)$ 的初始估计，其由前面的跟踪步骤得到。然后，根据在 3.6.4 节定义的 $\phi^{EW}(t+1)$ 或者 $\phi^{TW}(t+1)$ ，使其最小可以得到 $\mathbf{H}(t+1)$ 和 $\mathbf{H}^+(t+1)$ 的递归更新。最后，在式（3.6.8）中用 $\mathbf{H}^+(t+1)$ 代替 $\mathbf{H}^+(t)$ ，便得到 $\mathbf{b}(t+1)$ 的新的估计，然后将该新的行加到 $\mathbf{B}(t)$ 便得到 $\mathbf{B}(t+1)$ 。

2. PARAFAC-RLST 的步骤

① \mathbf{H} 的更新：给定 $\mathbf{b}(t+1)$ 的初始估计，导出 $\mathbf{H}(t+1)$ 的递归更新。令 $\nabla \phi(t+1) \in C^{IK \times R}$ 表示 $\phi(t+1)$ 关于 $\mathbf{H}(t+1)$ 的梯度。

- 在指数窗情况下：

$$\nabla \phi^{(EW)}(t+1) = 2 \sum_{\tau=1}^{t+1} \lambda^{t+1-\tau} (\mathbf{x}(\tau) - \mathbf{H}(t+1)\mathbf{b}^T(\tau)) \mathbf{b}^*(\tau) \quad (3.6.31)$$

- 在截断窗下：

$$\nabla \phi^{(TW)}(t+1) = 2 \sum_{\tau=1}^N \lambda^{N-\tau} (\mathbf{x}(u+\tau) - \mathbf{H}(t+1)\mathbf{b}^T(u+\tau)) \mathbf{b}^*(u+\tau) \quad (3.6.32)$$

解出 $\nabla \phi(t+1) = 0$ 。

- 在指数窗下：

$$\mathbf{H}(t+1) = \mathbf{R}_{EW}(t+1) \mathbf{P}_{EW}^{-1}(t+1) \quad (3.6.33)$$

其中

$$\begin{cases} \mathbf{R}_{EW}(t+1) = \sum_{\tau=1}^{t+1} \lambda^{t+1-\tau} \mathbf{x}(\tau) \mathbf{b}^*(\tau) \\ \mathbf{P}_{EW}(t+1) = \sum_{\tau=1}^{t+1} \lambda^{t+1-\tau} \mathbf{b}^T(\tau) \mathbf{b}^*(\tau) \end{cases}$$

• 在截断窗下:

$$\mathbf{H}(t+1) = \mathbf{R}_{TW}(t+1) \mathbf{P}_{TW}^{-1}(t+1) \quad (3.6.34)$$

其中

$$\begin{cases} \mathbf{R}_{TW}(t+1) = \sum_{\tau=1}^N \lambda^{N-\tau} \mathbf{x}(u+\tau) \mathbf{b}^*(u+\tau) \\ \mathbf{P}_{TW}(t+1) = \sum_{\tau=1}^N \lambda^{N-\tau} \mathbf{b}^T(u+\tau) \mathbf{b}^*(u+\tau) \end{cases} \quad (3.6.35)$$

\mathbf{R}_{EW} 、 \mathbf{P}_{EW} 、 \mathbf{R}_{TW} 、 \mathbf{P}_{TW} 的递归更新: 式 (3.6.36) 和式 (3.6.37)。

$$\begin{cases} \mathbf{R}_{EW}(t+1) = \lambda \mathbf{R}_{EW}(t) + \mathbf{x}(t+1) \mathbf{b}^*(t+1) \\ \mathbf{P}_{EW}(t+1) = \lambda \mathbf{P}_{EW}(t) + \mathbf{b}^T(t+1) \mathbf{b}^*(t+1) \end{cases} \quad (3.6.36)$$

$$\begin{cases} \mathbf{R}_{TW}(t+1) = \lambda \mathbf{R}_{TW}(t) + \mathbf{x}(t+1) \mathbf{b}^*(t+1) - \lambda^N \mathbf{x}(u) \mathbf{b}^*(u) \\ \mathbf{P}_{TW}(t+1) = \lambda \mathbf{P}_{TW}(t) + \mathbf{b}^T(t+1) \mathbf{b}^*(t+1) - \lambda^N \mathbf{b}^T(u) \mathbf{b}^*(u) \end{cases} \quad (3.6.37)$$

指数窗的特点是 \mathbf{R}_{EW} 、 \mathbf{P}_{EW} 的秩 1 更新结构时, 其中数据矩阵维数会增长。截断窗特点是 \mathbf{R}_{TW} 、 \mathbf{P}_{TW} 的秩 2 更新结构时, 旧的数据删除后会增加新的数据。这样式 (3.6.33) 中的矩阵 $\mathbf{Q}_{EW} \stackrel{\text{def}}{=} \mathbf{P}_{EW}^{-1}$ 就可以用递归的方式有效地计算出来, 根据矩阵求逆引理:

$$\mathbf{Q}_{EW}(t+1) = \lambda^{-1} \mathbf{Q}_{EW}(t) - \frac{\lambda^{-2} \mathbf{Q}_{EW}(t) \mathbf{b}^T(t+1) \mathbf{b}^*(t+1) \mathbf{Q}_{EW}(t)}{1 + \lambda^{-1} \mathbf{b}^*(t+1) \mathbf{Q}_{EW}(t) \mathbf{b}^T(t+1)} \quad (3.6.38)$$

同理, $\mathbf{Q}_{TW} \stackrel{\text{def}}{=} \mathbf{P}_{TW}^{-1}$ 可以通过两次求逆引理得到:

$$\mathbf{Q}_{TW}(t+1) = \tilde{\mathbf{Q}} + \frac{\lambda^N \tilde{\mathbf{Q}} \mathbf{b}^T(u) \mathbf{b}^*(u) \tilde{\mathbf{Q}}}{1 - \lambda^N \mathbf{b}^*(u) \tilde{\mathbf{Q}} \mathbf{b}^T(u)} \quad (3.6.39)$$

其中

$$\tilde{\mathbf{Q}} = \lambda^{-1} \mathbf{Q}_{TW}(t) - \frac{\lambda^{-2} \mathbf{Q}_{TW}(t) \mathbf{b}^T(t+1) \mathbf{b}^*(t+1) \mathbf{Q}_{TW}(t)}{1 + \lambda^{-1} \mathbf{b}^*(t+1) \mathbf{Q}_{TW}(t) \mathbf{b}^T(t+1)} \quad (3.6.40)$$

最后根据递归更新矩阵 $\mathbf{R}_{EW}(t+1)$ 、 $\mathbf{P}_{EW}^{-1}(t+1)$ 或者 $\mathbf{R}_{TW}(t+1)$ 、 $\mathbf{P}_{TW}^{-1}(t+1)$ 来更新式 (3.6.33) 或式 (3.6.34) 中的 $\mathbf{H}(t+1)$ 。

② \mathbf{H}^+ 的更新: 假设给定了以前的估计矩阵 $\mathbf{H}(t+1)$, 要估计 $\mathbf{B}(t+1)$ 的最后一行 $\mathbf{b}(t+1)$, 因为 $\mathbf{H}^+(t+1) = \mathbf{P}(t+1) \mathbf{R}^+(t+1)$, $\mathbf{b}(t+1)$ 的最小二乘估计如下。

在指数窗下:

$$\mathbf{b}^T(t+1) = \mathbf{P}_{EW}(t+1) \mathbf{R}_{EW}^+(t+1) \mathbf{x}(t+1) \quad (3.6.41)$$

在截断窗下:

$$\mathbf{b}^T(t+1) = \mathbf{P}_{TW}(t+1) \mathbf{R}_{TW}^+(t+1) \mathbf{x}(t+1) \quad (3.6.42)$$

现在目标就是找到 \mathbf{R}_{EW}^+ 和 \mathbf{R}_{TW}^+ 的递归更新:

$$\mathbf{R}_{EW}^+(t+1) = (\lambda \mathbf{R}_{EW}(t) + \mathbf{x}(t+1) \mathbf{b}^*(t+1))^+ \quad (3.6.43)$$

$$\mathbf{R}_{TW}^+(t+1) = (\lambda \mathbf{R}_{TW}(t) + \mathbf{x}(t+1)\mathbf{b}^*(t+1) - \lambda^N \mathbf{x}(u)\mathbf{b}^*(u))^+ \quad (3.6.44)$$

因为 $\mathbf{R}_{EW}^+(t+1)$ 有秩 1 更新结构, 所以通过伪逆引理, 可以被有效地计算出来。同样, 引理用两遍, 可以计算出 $\mathbf{R}_{TW}^+(t+1)$ 。

③ \mathbf{A} 、 \mathbf{B} 和 \mathbf{C} 的更新: 这些矩阵的更新类似于 PARAFAC-SDT。

3. 复杂度

表 3.6.3 总结了 PARAFAC-RLST 算法, 以及每一步对应的复杂度。截断窗时, 总复杂度是 $R^2(16IK + 72) + R(144IK + 10K + 20)$, 指数窗时, 是 $R^2(16KI + 40) + R(88KI + 10K + 10)$ 。对每一窗, PARAFAC-RLST 的复杂度都远低于 ALS。

表 3.6.3 PARAFAC-RLST 算法的总结^[19]

时刻 t 的输入: 旧估计值: $\mathbf{A}(t), \mathbf{B}(t), \mathbf{C}(t), \mathbf{P}(t), \mathbf{R}(t), \mathbf{P}^{-1}(t), \mathbf{R}^+(t)$ 观测值: $\mathbf{X}^{(1)}(t), \mathbf{X}^{(1)}(t+1) = \{\mathbf{X}^{(1)}(t), \mathbf{x}(t+1)\}$ 选择一个窗口 (EW 或 TW) 和 λ 在 $t+1$ 时刻更新		
步骤 1: 初始化 $\mathbf{b}^+(t+1)$ 的估计值		
更新		复杂度
$(EW): \mathbf{b}^+(t+1) = \mathbf{P}_{EW}(t)\mathbf{R}_{EW}^+(t)\mathbf{x}(t+1)$		$8RKI(R+1)$
$(TW): \mathbf{b}^+(t+1) = \mathbf{P}_{TW}(t)\mathbf{R}_{TW}^+(t)\mathbf{x}(t+1)$		$8RKI(R+1)$
步骤 2: 更新 \mathbf{P} 、 \mathbf{R} 、 \mathbf{P}^{-1} 、 \mathbf{R}^+		
更新		复杂度
步骤 2a: 更新 $\mathbf{R}_{EW}(t)$ 或 $\mathbf{R}_{TW}(t)$		
$(EW): \mathbf{R}_{EW}(t+1) = \lambda \mathbf{R}_{EW}(t) + \mathbf{x}(t+1)\mathbf{b}^*(t+1)$		$8RKI$
$(TW): \mathbf{R}_{TW}(t+1) = \lambda \mathbf{R}_{TW}(t) + \mathbf{x}(t+1)\mathbf{b}^*(t+1) - \lambda^N \mathbf{x}^T(u)\mathbf{b}^*(u)$		$16RKI$
步骤 2b: 更新 $\mathbf{P}_{EW}(t)$ 或 $\mathbf{P}_{TW}(t)$		
$(EW): \mathbf{P}_{EW}(t+1) = \lambda \mathbf{P}_{EW}(t) + \mathbf{b}^T(t+1)\mathbf{b}^*(t+1)$		$8R^2$
$(TW): \mathbf{P}_{TW}(t+1) = \lambda \mathbf{P}_{TW}(t) + \mathbf{b}^T(t+1)\mathbf{b}^*(t+1) - \lambda^N \mathbf{x}^T(u)\mathbf{b}^*(u)$		$16R^2$
步骤 2c: 更新 $\mathbf{R}_{EW}^+(t)$ 或 $\mathbf{R}_{TW}^+(t)$		
EW : 利用矩阵伪逆定理一次计算		$48RKI$
$(EW): \mathbf{R}_{EW}^+(t+1) = (\lambda \mathbf{R}_{EW}(t) + \mathbf{x}(t+1)\mathbf{b}^*(t+1))^+$		
TW : 利用矩阵伪逆定理二次计算		$96RKI$
$(TW): \mathbf{R}_{TW}^+(t+1) = (\lambda \mathbf{R}_{TW}(t) + \mathbf{x}(t+1)\mathbf{b}^*(t+1) - \lambda^N \mathbf{x}^T(u)\mathbf{b}^*(u))^+$		
步骤 2d: 更新 $\mathbf{P}_{EW}^{-1}(t)$ 或 $\mathbf{P}_{TW}^{-1}(t)$		
EW : 利用矩阵逆定理一次计算		$24R^2 + 10R$
$(EW): \mathbf{P}_{EW}^{-1}(t+1) = (\lambda \mathbf{P}_{EW}(t) + \mathbf{b}^T(t+1)\mathbf{b}^*(t+1))^{-1}$		$48R^2 + 20R$
TW : 利用矩阵逆定理二次计算		
$(TW): \mathbf{P}_{TW}^{-1}(t+1) = (\lambda \mathbf{P}_{TW}(t) + \mathbf{b}^T(t+1)\mathbf{b}^*(t+1) - \lambda^N \mathbf{x}^T(u)\mathbf{b}^*(u))^{-1}$		

续表

步骤 3: 更新 \mathbf{H} 、 \mathbf{A} 、 \mathbf{C}		
更新		复杂度
$EW: \mathbf{H}(t+1) = \mathbf{R}_{EW}(t+1)\mathbf{P}_{EW}^{-1}(t+1)$		$8R^2KI$
$TW: \mathbf{H}(t+1) = \mathbf{R}_{TW}(t+1)\mathbf{P}_{TW}^{-1}(t+1)$		$8R^2KI$
For $r = 1$ to R Do		$R \times$
$\mathbf{H}_r(t+1) = \text{unvec}([\mathbf{H}(t+1)_{:,r}])$		$8KI$
$\mathbf{a}_r^*(t+1) = \mathbf{H}_r^H(t+1)\mathbf{c}_r(t)$		
$\mathbf{c}_r(t+1) = \frac{\mathbf{H}_r(t+1)\mathbf{a}_r(t+1)}{\ \mathbf{H}_r(t+1)\mathbf{a}_r(t+1)\ }$		$8KI + 10K$
步骤 4: 更新 \mathbf{B}		
更新		复杂度
$EW: \mathbf{b}^T(t+1) = \mathbf{P}_{EW}(t+1)\mathbf{R}_{EW}^+(t+1)\mathbf{x}(t+1)$		$8R^2 + 8RKI$
$TW: \mathbf{b}^T(t+1) = \mathbf{P}_{TW}(t+1)\mathbf{R}_{TW}^+(t+1)\mathbf{x}(t+1)$		$8R^2 + 8RKI$
$\mathbf{B}^T(t+1) = [\mathbf{B}^T(t), \mathbf{b}^T(t+1)]$		
输出: 矩阵 $\mathbf{A}(t+1)$ 、 $\mathbf{B}(t+1)$ 、 $\mathbf{C}(t+1)$ 表示 $\mathcal{X}(t+1)$ PARAFAC 分解的估计值		

3.6.7 初始化

本小节讨论 PARAFAC-SDT 和 PARAFAC-RLST 在进入跟踪模式之前的初始化工作，并给出两种初始化方法。首先，用最开始的 J_{init} 个切片来构造张量，然后利用批处理算法来拟合 PARAFAC 模型。该初始化方法的处理延迟取决于该批处理算法的复杂度和收敛速度。

另一个观点就是构造基于特征值分解（Eigen Value Decomposition, EVD）的初始化方法，EVD 通常使用 ESPRIT-like 方法初始化批处理 PARAFAC 算法。考虑张量 \mathcal{X} 开始的两个观测切片构成，维数为 $I \times 2 \times K$ 。算法目标是找到 $\mathbf{A} \in C^{I \times R}$ 、 $\mathbf{C} \in C^{K \times R}$ 、 $\mathbf{B} \in C^{2 \times R}$ 的初始估计来拟合 \mathcal{X} 的 PARAFAC 模型。第一个 $I \times K$ 切片可以写成 $\mathbf{X}(1) = \mathbf{A}\mathbf{D}_1\mathbf{C}^T$ ，其中 $\mathbf{D}_1 = \text{diag}(\mathbf{b}_1)$ ， \mathbf{b}_1 是 \mathbf{B} 的第一行。同理 $\mathbf{X}(2) = \mathbf{A}\mathbf{D}_2\mathbf{C}^T$ 。假设 \mathbf{A} 和 \mathbf{C} 是列满秩的， \mathbf{D}_1 和 \mathbf{D}_2 是非奇异的， $\mathbf{D}_1\mathbf{D}_2^{-1}$ 的所有对角元素是不一样的。在这些条件下，PARAFAC 分解是唯一的。如果 \mathbf{A} 、 \mathbf{B} 、 \mathbf{C} 对模型拟合得很好（不存在噪声），可以通过下面这种简单且非迭代的方法得到上述矩阵。将这两个切面矩阵相结合，得到 $\mathbf{X}(1)\mathbf{X}^+(2) = \mathbf{A}\mathbf{D}_1\mathbf{D}_2^{-1}\mathbf{A}^+$ ， \mathbf{A} 可以通过 $\mathbf{X}(1)\mathbf{X}^+(2)$ 的 R 个特征向量估计出来。同理， \mathbf{C} 也可以从 $\mathbf{X}^+(1)\mathbf{X}(2)$ 的 EVD 分解得到。给定 \mathbf{A} 、 \mathbf{C} 、 \mathbf{B} 可以以最小二乘估计出来。存在噪声时，虽然通过上述方法得到的矩阵可能并不能很好地对模型进行拟合，但是其可以以较低的复杂度为批处理提供较好的初始值。

本节中研究了两种自适应算法，来跟踪三阶张量的 PARAFAC 分解。第一种算法称为 PARAFAC-SDT，是基于同时对角化的批处理算法的跟踪版本。第二种算法称为 PARAFAC-RLST，是基于加权最小二乘最小化准则。

3.7 大规模 PARAFAC 分解

对于大规模 PARAFAC 分解, 标准 ALS 算法涉及 Khatri-Rao 积运算, 消耗大量的内存, 导致难以实现。如何进行大规模 PARAFAC 分解, 这是个难题。文献[24]提出了一些算法用于大规模 PARAFAC 分解。

3.7.1 张量符号与基本模型

定义: 采用有下画线的加粗大写字母定义一个张量, 如 $\underline{\mathbf{Y}}$, 加粗大写字母表示矩阵, 如 \mathbf{A} , 加粗小写字母表示向量, 如 \mathbf{d} 。对于一个三维张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I \times J \times K}$, 它的前向切片、侧向切片、水平切片分别表示为 $\mathbf{Y}_k = \mathbf{Y}_{:,k}$ 、 $\mathbf{Y}_{:,j}$ 、 $\mathbf{Y}_{i,:}$ 。沿着模 3 在位置 (i,j) 处的向量表示为 \mathbf{y}_{ij} , 相应地, 模 2 和模 1 在该位置的向量分别表示为 $\mathbf{y}_{i,k}$ 和 $\mathbf{y}_{j,k}$ 。张量 $\underline{\mathbf{Y}}$ 的模 n 矩阵形式表示为 $\mathbf{Y}_{(n)}$, 一个 3 维张量的模 n 矩阵形式可以直观地定义为

$$\mathbf{Y}_{(1)} = [\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_K] \in \mathbb{R}^{I \times JK} \quad (3.7.1)$$

$$\mathbf{Y}_{(2)} = [\mathbf{Y}_1^T, \mathbf{Y}_2^T, \dots, \mathbf{Y}_K^T] \in \mathbb{R}^{J \times IK} \quad (3.7.2)$$

$$\mathbf{Y}_{(3)} = [\text{vec}(\mathbf{Y}_1), \text{vec}(\mathbf{Y}_2), \dots, \text{vec}(\mathbf{Y}_K)]^T \in \mathbb{R}^{K \times IJ} \quad (3.7.3)$$

一个张量与一个矩阵的 n 模乘积表示为

$$\underline{\mathbf{Y}} = \mathbf{G} \times_n \mathbf{A} \quad \text{或} \quad \mathbf{Y}_{(n)} = \mathbf{A} \mathbf{G}_{(n)} \quad (3.7.4)$$

一个张量 $\underline{\mathbf{G}}$ 和一组矩阵集 $\mathbf{A}^{(n)}$ 的所有可能的 n ($n=1, 2, \dots, N$) 模乘积可以表示为

$$\underline{\mathbf{G}} \times \{\mathbf{A}\} = \underline{\mathbf{G}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times \dots \times_N \mathbf{A}^{(N)} \quad (3.7.5)$$

$$[\underline{\mathbf{G}} \times \{\mathbf{A}\}]_{(n)} = \mathbf{A}^{(n)} \mathbf{G}_{(n)} [\mathbf{A}^{(1)} \otimes \dots \otimes \mathbf{A}^{(n-1)} \otimes \mathbf{A}^{(n+1)} \otimes \dots \otimes \mathbf{A}^{(N)}] \quad (3.7.6)$$

上述中除了模 n 的模积可以表示为

$$[\underline{\mathbf{G}} \times_n \{\mathbf{A}\}] = \underline{\mathbf{G}} \times_1 \mathbf{A}^{(1)} \times \dots \times_2 \mathbf{A}^{(2)} \times \dots \times_{n-1} \mathbf{A}^{(n-1)} \times_{n+1} \mathbf{A}^{(n+1)} \times \dots \times_N \mathbf{A}^{(N)} \quad (3.7.7)$$

一个张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ 和一个向量 $\mathbf{a} \in \mathbb{R}^{I_n}$ 的 n 模乘积可以表示为

$$\underline{\mathbf{Z}} = \underline{\mathbf{Y}} \times_n \mathbf{a} \in \mathbb{R}^{I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N} \quad (3.7.8)$$

一个张量 $\underline{\mathbf{Y}}$ 和一个 N 列向量集 $\{\mathbf{a}\} = [\mathbf{a}^{(1)}, \mathbf{a}^{(2)}, \dots, \mathbf{a}^{(N)}]$ 的张量-向量积可以表示为

$$\underline{\mathbf{Y}} \times \{\mathbf{a}\} = \underline{\mathbf{Y}} \times_1 \mathbf{a}^{(1)} \times_2 \mathbf{a}^{(2)} \times \dots \times_N \mathbf{a}^{(N)} \quad (3.7.9)$$

PARAFAC^[5]可以阐述为将一个给定的 N 阶数据张量 $\underline{\mathbf{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ 分解成 N 个矩阵分量: $\mathbf{A}^{(n)} = [\mathbf{a}_1^{(n)}, \mathbf{a}_2^{(n)}, \dots, \mathbf{a}_J^{(n)}] \in \mathbb{R}^{I_n \times J}$, ($n=1, 2, \dots, N$) 表示公 (承载) 因子, 即

$$\underline{\mathbf{Y}} \approx \sum_{j=1}^J \mathbf{a}_j^{(1)} \circ \mathbf{a}_j^{(2)} \circ \dots \circ \mathbf{a}_j^{(N)} \quad (3.7.10)$$

$$= [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \dots, \mathbf{A}^{(N)}] = [\{\mathbf{A}\}] = \hat{\underline{\mathbf{Y}}}$$

其中, 符号 \circ 表示外积, 假设单位长因子 $\|\mathbf{a}_j^{(n)}\|_p = 1$, $n=1, 2, \dots, N-1$, $j=1, 2, \dots, J$, $p=1, 2$ (图

3.7.1)。张量 $\hat{\underline{Y}}$ 是数据张量 \underline{Y} 的估计值。此外，也可以使用张量符号来描述 PARAFAC 模型，即

$$\underline{Y} = \underline{I} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times \cdots \times_N \mathbf{A}^{(N)} \quad (3.7.11)$$

其中 \underline{I} 是一个单位张量。式 (3.7.11) 中的张量的模- n 矩阵化后可以由一系列矩阵因子表示：

$$\begin{aligned} \underline{Y}_{(n)} &\approx \mathbf{A}^{(n)} (\mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)} \odot \mathbf{A}^{(n-1)} \cdots \odot \mathbf{A}^{(1)})^T, \quad n=1, 2, \dots, N-1 \\ &= \mathbf{A}^{(n)} (\mathbf{A})^{\odot_{-n} T} \end{aligned} \quad (3.7.12)$$

著名的 PARAFAC 算法是依次 F 范数最小的交替最小二乘 (ALS) 算法。

$$D(\underline{Y} \parallel \hat{\underline{Y}}) = \frac{1}{2} \|\underline{Y} - \hat{\underline{Y}}\|_F^2 \quad (3.7.13)$$

通过对因子 $\mathbf{A}^{(n)}$ 用更新法则：

$$\begin{aligned} \mathbf{A}^{(n)} &\leftarrow \underline{Y}_{(n)} \{\mathbf{A}\}^{\odot_{-n}} (\{\mathbf{A}\}^{\odot_{-n} T} \{\mathbf{A}\}^{\odot_{-n}})^{-1}, \quad n=1, 2, \dots, N \\ &= \underline{Y}_{(n)} \{\mathbf{A}\}^{\odot_{-n}} (\{\mathbf{A}^T \mathbf{A}\}^{\oplus_{-n}})^{-1} \end{aligned} \quad (3.7.14)$$

其中 $\underline{Y}_{(n)}$ 是张量 \underline{Y} 的模 n 矩阵化形式，Hadamard 积为

$$\{\mathbf{A}\}^{\oplus_{-n}} = \mathbf{A}^{(N)} \oplus \cdots \oplus \mathbf{A}^{(n+1)} \oplus \mathbf{A}^{(n-1)} \cdots \oplus \mathbf{A}^{(1)}$$

对于大规模张量，大小为 $I_n \times (\prod_{k \neq n} I_k)$ 的模 n 矩阵化形式 $\underline{Y}_{(n)}$ 通常是一个大矩阵，且 Khatri-Rao 积 $\{\mathbf{A}\}^{\odot_{-n}}$ 在每次迭代后返回一个大小为 $(\prod_{k \neq n} I_k) \times J$ 的高矩阵。因此，这种学习准则需要很高的计算复杂度，在大规模数据下也需要耗费大量的内存。标准 ALS 算法不能分解一个包含成千上万个元素的密集数据张量。处理这种大数据集，可以通过沿着每个满足特定准则^[24,25]的模 n 采样向量，以减少 $\underline{Y}_{(n)}$ 的列数来解决，而不是计算式 (3.7.14) 中的整个张量。块结构和 CUR^[26]方法都不能完全解决超大规模数据问题，因为它们始终需要计算 Khatri-Rao 积。

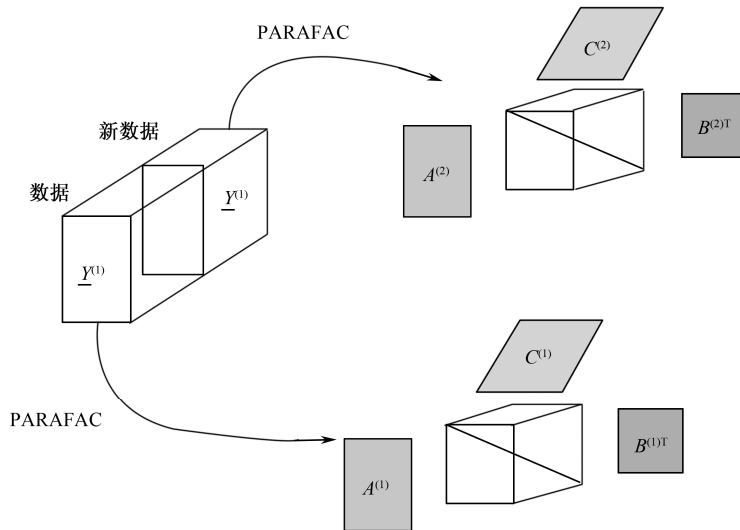


图 3.7.1 一个张量自动分解为两个子张量的图解^[24]

3.7.2 动态张量分解

为了简要说明基本概念，将首先考虑三维张量分解为两个子张量的网络 PARAFAC。这种模型可以认为是动态张量分析的一种简单模式，其数据沿着单一模式扩充。

问题 1（动态张量分解）：

考虑两个数据张量 $\underline{Y}^{(1)} \in \mathbf{R}^{I \times J \times K_1}$ ， $\underline{Y}^{(2)} \in \mathbf{R}^{I \times J \times K_2}$ 以及它们近似的 PARAFAC 模型

$$\underline{Y}^{(k)} \approx \underline{I} \times_1 \mathbf{A}^{(k)} \times_2 \mathbf{B}^{(k)} \times_3 \mathbf{C}^{(k)}, \quad k=1,2 \quad (3.7.15)$$

其中 $\mathbf{A}^{(k)} \in \mathbf{R}^{I \times R_k}$ ， $\mathbf{B}^{(k)} \in \mathbf{R}^{J \times R_k}$ 和 $\mathbf{C}^{(k)} \in \mathbf{R}^{K_k \times R_k}$ 。

问题是找到连续张量 $\underline{Y} \in \mathbf{R}^{I \times J \times K}$ 的 PARAFAC 因子（其中 $K = K_1 + K_2$ ）：

$$\underline{Y} \approx \underline{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \quad (3.7.16)$$

解决这个问题的最简单方法是分解合成张量。这个方法提供准确的结果。然而，对于大规模数据，较高的计算复杂度使得此解决方法不合适。本节将给出三种方法来解决这个问题：

- ① 最小化合成张量的代价函数；
- ② 采用 ALS 算法或改进算法进行张量分解；
- ③ 合成子张量的因子。

所有方法的最终学习准则是统一的。

1. 方法 1：直接最小化代价函数

假设合成张量 \underline{Y} 可以被因式分解为 R 部分： $\mathbf{A} \in \mathbf{R}^{I \times R}$ ， $\mathbf{B} \in \mathbf{R}^{J \times R}$ 和 $\mathbf{C} \in \mathbf{R}^{K \times R}$ 。如果将 \mathbf{C} 分为两个矩阵 $\mathbf{C} = [\mathbf{C}_{(1)}^T, \mathbf{C}_{(2)}^T]^T$ ， $\mathbf{C}_{(1)} \in \mathbf{R}^{K_1 \times R}$ 和 $\mathbf{C}_{(2)} \in \mathbf{R}^{K_2 \times R}$ ，子张量 $\underline{Y}^{(1)}$ 和 $\underline{Y}^{(2)}$ 可以通过它们的公因子近似得到：

$$\underline{Y}^{(1)} \approx \underline{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}_{(1)} \quad (3.7.17)$$

$$\underline{Y}^{(2)} \approx \underline{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}_{(2)} \quad (3.7.18)$$

通常，一个张量分解可以通过最小化合成张量的代价函数来解决，它的近似为

$$\begin{aligned} D &= \frac{1}{2} \|\underline{Y} - \underline{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}\|_F^2 \\ &= \frac{1}{2} \sum_{k=1}^2 \|\underline{Y}^{(k)} - \underline{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}_{(k)}\|_F^2 \\ &= \frac{1}{2} \sum_{k=1}^2 \|\underline{Y}^{(k)} - \mathbf{A}(\mathbf{C}_{(k)} \odot \mathbf{B})^T\|_F^2 \end{aligned} \quad (3.7.19)$$

为了找到因子 \mathbf{A} 的学习准则，将代价函数写为式 (3.7.19) 中的模 1 矩阵化，计算它对 \mathbf{A} 的偏导：

$$\begin{aligned}
 \nabla_A D &= \sum_{k=1}^2 (-Y_{(1)}^{(k)} (C_{(k)} \odot B) + A(C_{(k)} \odot B)^T (C_{(k)} \odot B)) \\
 &= \sum_{k=1}^2 (-Y_{(1)}^{(k)} (C_{(k)} \odot B) + A(C_{(k)}^T C_{(k)}) \oplus (B^T B)) \\
 &= \sum_{k=1}^2 -Y_{(1)}^{(k)} (C_{(k)} \odot B) + A(C^T C) \oplus (B^T B)
 \end{aligned} \tag{3.7.20}$$

将式 (3.7.20) 中的 $\underline{Y}^{(1)}$ 和 $\underline{Y}^{(2)}$ 用式 (3.7.15) 的近似值代替, 并将式 (3.7.20) 置 0, 可得到矩阵 A 的一个更新准则:

$$\begin{aligned}
 A &\leftarrow \left(\sum_{k=1}^2 Y_{(1)}^{(k)} (C_{(k)} \odot B) \right) ((C^T C) \oplus (B^T B))^{-1} \\
 &= \left(\sum_{k=1}^2 A^{(k)} (C^{(k)} \odot B^{(k)})^T (C_{(k)} \odot B) \right) ((C^T C) \oplus (B^T B))^{-1} \\
 &= \left(\sum_{k=1}^2 A^{(k)} (C^{(k)T} C_{(k)}) (B^{(k)T} B) \right) ((C^T C) \oplus (B^T B))^{-1} \\
 &= \left(\sum_{k=1}^2 A^{(k)} (P^{(k)} ./ (A^{(k)T} A)) \right) (Q ./ (A^T A))^{-1}
 \end{aligned} \tag{3.7.21}$$

其中, “./” 为矩阵点除。

$$P^{(k)} = (A^{(k)T} A) \oplus (B^{(k)T} B) \oplus (C^{(k)T} C) \tag{3.7.22}$$

$$Q = (A^T A) \oplus (B^T B) \oplus (C^T C) \tag{3.7.23}$$

相似地, 矩阵 B 的学习准则为

$$B \leftarrow \left(\sum_{k=1}^2 B^{(k)} (P^{(k)} ./ (B^{(k)T} B)) \right) (Q ./ (B^T B))^{-1} \tag{3.7.24}$$

因子 C 采用部分更新 $C_{(1)}$ 和 $C_{(2)}$ 的方式得到。从张量 $\underline{Y}^{(k)}$ 更新 $C_{(k)}$ ($k=1, 2$) 的修正 ALS 学习准则为

$$\begin{aligned}
 C_{(k)} &\leftarrow Y_{(3)}^{(k)} (B \odot A) ((B^T B) \oplus (A^T A))^{-1} \\
 &= C^{(k)} (B^{(k)} \odot A^{(k)})^T (B \odot A) ((B^T B) \oplus (A^T A))^{-1} \\
 &= C^{(k)} ((B^{(k)T} B) \oplus (A^{(k)T} A)) ((B^T B) \oplus (A^T A))^{-1} \\
 &= C^{(k)} (P^{(k)} ./ (C^{(k)T} C)) (Q ./ (C^T C))^{-1}
 \end{aligned} \tag{3.7.25}$$

注意到矩阵 $P^{(k)} \in R^{R_k \times R}$ 和 $Q \in R^{R \times R}$ 维度相当小。因此, 式 (3.7.21)、式 (3.7.24) 和式 (3.7.25) 都是低复杂度的^[24]。

2. 方法 2: 修改 ALS 学习准则

对于合成张量 \underline{Y} , 因子 A 、 B 和 C 可以通过式 (3.7.14) 的 ALS 算法估计。

$$A \leftarrow Y_{(1)} (C \odot B) ((C^T C) \oplus (B^T B))^{-1} \tag{3.7.26}$$

通过采用下面的关系, 可以将 ALS 学习准则转换为式 (3.7.21) 的学习准则。

$$\begin{aligned}
Y_{(1)}(\mathbf{C} \odot \mathbf{B}) &= [\mathbf{Y}_{(1)}^{(1)}, \mathbf{Y}_{(1)}^{(2)}] \left[\begin{bmatrix} \mathbf{C}^{(1)} \\ \mathbf{C}^{(2)} \end{bmatrix} \odot \mathbf{B} \right] \\
&= [\mathbf{Y}_{(1)}^{(1)}, \mathbf{Y}_{(1)}^{(2)}] \begin{bmatrix} \mathbf{C}^{(1)} \odot \mathbf{B} \\ \mathbf{C}^{(2)} \odot \mathbf{B} \end{bmatrix} = \sum_{k=1}^2 \mathbf{Y}_{(1)}^{(k)} (\mathbf{C}^{(k)} \odot \mathbf{B}) \\
&= \sum_{k=1}^2 \mathbf{A}^{(k)} (\mathbf{C}^{(k)} \odot \mathbf{B}^{(k)})^T (\mathbf{C}^{(k)} \odot \mathbf{B}) \\
&= \sum_{k=1}^2 \mathbf{A}^{(k)} ((\mathbf{C}^{(k)T} \mathbf{C}^{(k)}) \oplus (\mathbf{B}^{(k)T} \mathbf{B}))
\end{aligned} \tag{3.7.27}$$

3. 方法 3: 合成因子

定理 3.7.1 (合成因子): 张量 $\underline{\mathbf{Y}}$ 被分解为若干因子 (每个由 $\tilde{\mathbf{R}} = \mathbf{R}_1 + \mathbf{R}_2$ 部分组成):

$$\tilde{\mathbf{A}} = [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}] \tag{3.7.28}$$

$$\tilde{\mathbf{B}} = [\mathbf{B}^{(1)}, \mathbf{B}^{(2)}] \tag{3.7.29}$$

$$\tilde{\mathbf{C}} = \begin{bmatrix} \mathbf{C}^{(1)} \\ \mathbf{C}^{(2)} \end{bmatrix} \tag{3.7.30}$$

则有

$$\underline{\mathbf{Y}} \approx \underline{\mathbf{I}} \times_1 \tilde{\mathbf{A}} \times_2 \tilde{\mathbf{B}} \times_3 \tilde{\mathbf{C}} \tag{3.7.31}$$

证明: 考虑合成张量 $\underline{\mathbf{Y}}$ 的模 1 的矩阵化形式:

$$\begin{aligned}
\mathbf{Y}_{(1)} &= [\mathbf{Y}_{(1)}^{(1)}, \mathbf{Y}_{(1)}^{(2)}] \\
&= [\mathbf{A}^{(1)} (\mathbf{C}^{(1)} \odot \mathbf{B}^{(1)})^T, \mathbf{A}^{(2)} (\mathbf{C}^{(2)} \odot \mathbf{B}^{(2)})^T] \\
&= [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}] \begin{bmatrix} (\mathbf{C}^{(1)} \odot \mathbf{B}^{(1)})^T \\ (\mathbf{C}^{(2)} \odot \mathbf{B}^{(2)})^T \end{bmatrix} \\
&= [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}] \left[\begin{bmatrix} \mathbf{C}^{(1)} \\ 0 \end{bmatrix} \odot \mathbf{B}^{(1)}, \begin{bmatrix} 0 \\ \mathbf{C}^{(2)} \end{bmatrix} \odot \mathbf{B}^{(2)} \right]^T \\
&= [\mathbf{A}^{(1)}, \mathbf{A}^{(2)}] \begin{bmatrix} \mathbf{C}^{(1)} \\ \mathbf{C}^{(2)} \end{bmatrix} \odot [\mathbf{B}^{(1)}, \mathbf{B}^{(2)}]^T \\
&= \tilde{\mathbf{A}} (\tilde{\mathbf{C}} \odot \tilde{\mathbf{B}})^T
\end{aligned} \tag{3.7.32}$$

由此得到张量 $\underline{\mathbf{Y}}$ 关于 $\tilde{\mathbf{A}}$ 、 $\tilde{\mathbf{B}}$ 、 $\tilde{\mathbf{C}}$ 的因式分解为

$$\underline{\mathbf{Y}} \approx \underline{\mathbf{I}} \times_1 \tilde{\mathbf{A}} \times_2 \tilde{\mathbf{B}} \times_3 \tilde{\mathbf{C}} \tag{3.7.33}$$

证毕。

定理 3.7.1 保证了任何大规模的张量都可以被因式分解为子张量的级联因子。然而, 组合因子通常有更高的秩。接下来的更新准则将有助于根据因子 $\tilde{\mathbf{A}}$ 、 $\tilde{\mathbf{B}}$ 、 $\tilde{\mathbf{C}}$ 找到 R 个更小的因子。

因子 \mathbf{A} 可以通过对张量 $\underline{\mathbf{Y}}$ 采用式 (3.7.14) 中改进的 ALS 学习准则更新^[24]:

$$\begin{aligned}
 A &\leftarrow Y_{(1)}(C \odot B)((C^T C) \oplus (B^T B))^{-1} \\
 &= \tilde{A}(\tilde{C} \odot \tilde{B})^T(C \odot B)((C^T C) \oplus (B^T B))^{-1} \\
 &= \tilde{A}((\tilde{C}^T C) \oplus (\tilde{B}^T B))((C^T C) \oplus (B^T B))^{-1} \\
 &= \tilde{A}(P./(\tilde{A}^T A)(Q./(\tilde{A}^T A))^{-1}
 \end{aligned} \tag{3.7.34}$$

其中

$$P = (\tilde{A}^T A) \oplus (\tilde{B}^T B) \oplus (C^T C) \tag{3.7.35}$$

注意，由于下面的关系，式 (3.7.21) 和式 (3.7.34) 的学习准则是等价的。

$$\begin{aligned}
 \tilde{C}^T C &= \begin{bmatrix} \tilde{C}^{(1)} \\ \tilde{C}^{(2)} \end{bmatrix}^T \begin{bmatrix} C^{(1)} \\ C^{(2)} \end{bmatrix} = \begin{bmatrix} \tilde{C}^{(1)T} C_{(1)} \\ \tilde{C}^{(2)T} C_{(1)} \end{bmatrix} \\
 \tilde{B}^T B &= [\tilde{B}^{(1)}, \tilde{B}^{(2)}]^T B = \begin{bmatrix} \tilde{B}^{(1)T} B \\ \tilde{B}^{(2)T} B \end{bmatrix}
 \end{aligned}$$

和

$$\tilde{A}((\tilde{C}^T C) \oplus (\tilde{B}^T B)) = \sum_{k=1}^2 A^{(k)}(\tilde{C}^{(k)T} C_{(k)}) \oplus (\tilde{B}^{(k)T} B)$$

3.7.3 网格 PARAFAC

考虑一个张量 \underline{Y} 被分成大小为 $I_{k_1} \times I_{k_2} \times \cdots \times I_{k_N}$ ($\sum_{k_n=1}^{K_n} I_{k_n} = I_n$) 的多个子张量 $\underline{Y}^{(k)}$ 网格，其中向量 $\bar{k} = [k_1, k_2, \dots, k_N]$ 为子张量的索引， $1 \leq k_n \leq K_n$ ， K_n 是沿着模 n 的子张量个数(图 3.7.2)。通过并行进行子张量分解获得子因子 $U_{(\bar{k})}^{(n)}$ 。最终，整个张量的完整因子 $A^{(n)}$ 可以通过这些快速学习准则和并行计算的子因子估计得到^[24]。

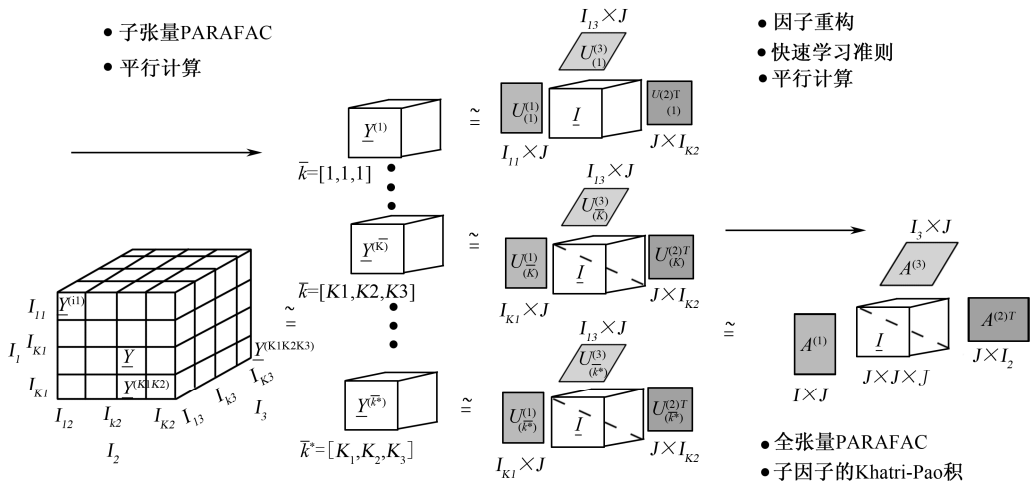


图 3.7.2 对于大规模张量的标准 PARAFAC 分解和网格 PARAFAC 分解两个步骤的图例^[24]

1. 网格 PARAFAC 下的 ALS 算法

假设数据张量 \underline{Y} 可以通过 N 个因子 $\mathbf{A}^{(n)}$ 近似计算得到, 将每个因子 $\mathbf{A}^{(n)}$ 分成 K_n 部分:

$$\mathbf{A}^{(n)} = [\mathbf{A}_{(1)}^{(n)\top}, \mathbf{A}_{(2)}^{(n)\top}, \dots, \mathbf{A}_{(K_n)}^{(n)\top}]^\top \quad (3.7.36)$$

其中子因子 $\mathbf{A}_{(k_n)}^{(n)} \in \mathbf{R}^{I_{k_n} \times J}$, $k_n = 1, 2, \dots, K_n$ 。

定理 3.7.2: 一个子张量 $\underline{Y}^{(\bar{k})}$, $\bar{k} = [k_1, k_2, \dots, k_N]$ 可以被分解为一组 N 个子因子 $\{\mathbf{A}_{(\bar{k})}\} = \{\mathbf{A}_{(k_1)}^{(1)}, \mathbf{A}_{(k_2)}^{(2)}, \dots, \mathbf{A}_{(k_N)}^{(N)}\}$:

$$\begin{aligned} \underline{Y}^{(\bar{k})} &\approx \underline{I} \times_1 \mathbf{A}_{(k_1)}^{(1)} \times_2 \mathbf{A}_{(k_2)}^{(2)} \times \dots \times_N \mathbf{A}_{(k_N)}^{(N)} \\ &= \llbracket \{\mathbf{A}_{(\bar{k})}\} \rrbracket \end{aligned} \quad (3.7.37)$$

证明: 这个定理的证明可以直接从 PARAFAC 模型对每个元素 $y_{\bar{i}}$ 的定义得到,

$$\begin{aligned} \bar{i} &= [i_1, i_2, \dots, i_N], \quad \sum_{l=1}^{k_n-1} I_{k_l} < i_n < \sum_{l=1}^{k_n} I_{k_l}, \\ y_{\bar{i}} &= a_{i_1}^{(1)} a_{i_2}^{(2)} \dots a_{i_N}^{(N)} \end{aligned} \quad (3.7.38)$$

基于网格 PARAFAC 的 ALS 算法最小化所有子张量的欧氏距离:

$$D = \frac{1}{2} \left\| \underline{Y} - \underline{I} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times \dots \times_N \mathbf{A}^{(N)} \right\|_F^2 \quad (3.7.39)$$

$$\begin{aligned} &= \frac{1}{2} \sum_{k_1=1}^{K_1} \dots \sum_{K_N=1}^{K_N} \left\| \underline{Y}^{(\bar{k})} - \llbracket \{\mathbf{A}_{(\bar{k})}\} \rrbracket \right\|_F^2 \\ &= \frac{1}{2} \sum_{\bar{k}} \left\| \underline{Y}_{(n)}^{(\bar{k})} - \mathbf{A}_{(k_n)}^{(n)} \{\mathbf{A}_{(\bar{k})}\}^{\odot_{-n} T} \right\|_F^2 \end{aligned} \quad (3.7.40)$$

子因子 $\mathbf{A}_{(k_n)}^{(n)}$ 的偏导为

$$\begin{aligned} \nabla_{\mathbf{A}_{(k_n)}^{(n)}} D &= \sum_{k_n=k_n} (-\underline{Y}_{(n)}^{(\bar{k})} \mathbf{A}_{(\bar{k})}^{\odot_{-n}} + \mathbf{A}_{(k_n)}^{(n)} \mathbf{A}_{(\bar{k})}^{\odot_{-n} T} \mathbf{A}_{(\bar{k})}^{\odot_{-n}}) \\ &= \sum_{k_n=k_n} (-\underline{Y}_{(n)}^{(\bar{k})} \mathbf{A}_{(\bar{k})}^{\odot_{-n}} + \mathbf{A}_{(k_n)}^{(n)} \{\mathbf{A}_{(\bar{k})}\}^{\oplus_{-n}}) \end{aligned} \quad (3.7.41)$$

由此得到子因子 $\mathbf{A}_{(k_n)}^{(n)}$ 的学习准则为

$$\mathbf{A}_{(k_n)}^{(n)} \leftarrow \left(\sum_{k_n=k_n} (\underline{Y}_{(n)}^{(\bar{k})} \mathbf{A}_{(\bar{k})}^{\odot_{-n}}) \right) \left(\sum_{k_n=k_n} (\mathbf{A}_{(\bar{k})}^{\oplus_{-n}} \mathbf{A}_{(\bar{k})}^{\odot_{-n}}) \right)^{-1} \quad (3.7.42)$$

由于子因子的大小相对较小, $\underline{Y}_{(n)}^{(\bar{k})} \mathbf{A}_{(\bar{k})}^{\odot_{-n}}$ 、 $(\mathbf{A}_{(\bar{k})}^{\oplus_{-n}} \mathbf{A}_{(\bar{k})}^{\odot_{-n}})$ 在并行或顺序执行的计算机上可以很快地实现。而且, 可以消除包含子因子 $\mathbf{A}_{(k_n)}^{(n)}$ 估计值的子张量, 得到 CUR 分解抽取的列向量。

下面给出一个可以避免式 (3.7.42) 中 Khatri-Rao 积计算的最优化算法。

2. 最优 ALS 学习准则

对于子张量 $\underline{Y}^{(\bar{k})}$, 采用式 (3.7.14) 中 PARAFAC 的 ALS 算法将张量分解为 J_k 部分。

$$\underline{Y}^{(\bar{k})} \approx \underline{I} \times_1 \mathbf{U}_{(\bar{k})}^{(1)} \times_2 \mathbf{U}_{(\bar{k})}^{(2)} \times \dots \times_N \mathbf{U}_{(\bar{k})}^{(N)} \quad (3.7.43)$$

应该选择秩-1 张量数 $J_{\bar{k}}$ ，使得因子 $U_{(\bar{k})}^{(n)}$ 尽可能地解释子张量 $\underline{Y}^{(\bar{k})}$ 。因为子张量 $\underline{Y}^{(\bar{k})}$ 维度小，这个因式分解可以很容易实现。对于一个子张量，有

$$\begin{aligned}\underline{Y}_{(n)}^{(\bar{k})} \mathbf{A}_{(\bar{k})}^{\odot -n} &\approx \mathbf{U}_{(\bar{k})}^{(n)} \mathbf{U}_{(\bar{k})}^{\odot -n \top} \mathbf{A}_{(\bar{k})}^{\odot -n} = \mathbf{U}_{(\bar{k})}^{(n)} (\mathbf{U}_{(\bar{k})}^{\top} \mathbf{A}_{(\bar{k})})^{\oplus -n} \\ &= \mathbf{U}_{(\bar{k})}^{(n)} (\mathbf{P}_{(\bar{k})} ./ (\mathbf{U}_{(\bar{k})}^{(n)\top} \mathbf{A}_{(k_n)}^n))\end{aligned}\quad (3.7.44)$$

其中， $\mathbf{P}_{(\bar{k})} = (\mathbf{U}_{(\bar{k})}^{\top} \mathbf{A}_{(\bar{k})})^{\oplus} \in \mathbf{R}^{J_{\bar{k}} \times J}$ ，令 $\mathbf{Q}_{(\bar{k})} = (\mathbf{A}_{(\bar{k})}^{\top} \mathbf{A}_{(\bar{k})})^{\oplus} \in \mathbf{R}^{J \times J}$ 。由式 (3.7.42) 和式 (3.7.44)，得到子因子 $\mathbf{A}_{(k_n)}^{(n)}$ 的快速更新准则：

$$\mathbf{A}_{(k_n)}^{(n)} \leftarrow \left(\sum_{\bar{k}_n=k_n} \mathbf{U}_{(\bar{k})}^{(n)} (\mathbf{P}_{(\bar{k})} ./ (\mathbf{U}_{(\bar{k})}^{(n)\top} \mathbf{A}_{(k_n)}^n)) \right) \quad (3.7.45)$$

$$\left(\sum_{\bar{k}_n=k_n} (\mathbf{Q}_{(\bar{k})} ./ (\mathbf{A}_{(k_n)}^{(n)\top} \mathbf{A}_{(k_n)}^n)) \right)^{-1} = \mathbf{T} \mathbf{S}^{-1} \quad (3.7.46)$$

其中矩阵 \mathbf{T} 和 \mathbf{S} 由具体的子因子 $\mathbf{A}_{(k_n)}^{(n)}$ 计算得到：

$$\mathbf{T} = \sum_{\substack{k_1, \dots, k_{n-1} \\ k_{n+1}, \dots, k_N}} \mathbf{U}_{(\bar{k})}^{(n)} (\mathbf{P}_{(\bar{k})} ./ (\mathbf{U}_{(\bar{k})}^{(n)\top} \mathbf{A}_{(k_n)}^n)) \quad (3.7.47)$$

$$\mathbf{S} = \sum_{\substack{k_1, \dots, k_{n-1} \\ k_{n+1}, \dots, k_N}} (\mathbf{Q}_{(\bar{k})} ./ (\mathbf{A}_{(k_n)}^{(n)\top} \mathbf{A}_{(k_n)}^n)) \quad (3.7.48)$$

式 (3.7.47) 和式 (3.7.48) 中两个求和的每一项是计算 Hadamard 除法，并执行小矩阵的所有运算，而不是所有高矩阵的 Khatri-Rao 积。矩阵 $\mathbf{P}_{(\bar{k})} \in \mathbf{R}^{J_{\bar{k}} \times J}$ 和 $\mathbf{Q}_{(\bar{k})} \in \mathbf{R}^{J \times J}$ 可以只计算一次，然后在得到子因子 $\mathbf{A}_{(k_n)}^{(n)}$ 的估计后可以很快地被更新。而且，矩阵 \mathbf{T} 和 \mathbf{S} 可以通过并行循环计算。

新 ALS 算法的伪代码在算法 3.7.1 中给出，用 “parfor” 表示并行 for 循环。

算法 3.7.1 大规模 PAFAC 的快速 ALS 算法^[24]

输入：大小为 $I_1 \times I_2 \times \dots \times I_N$ 的数据 \underline{Y} ，基本组成部分数 J 。

输出：使得式 (3.7.40) 的代价函数最小的 N 个因子 $\mathbf{A}^{(n)} \in \mathbf{R}^{I_n \times J}$ 。

begin

 初始化 $\mathbf{A}_{(k_n)}^{(n)}$ ，任取 n, k_n ；

parfor 子张量 $\underline{Y}^{(k)}$ ，do

$[\mathbf{U}_{(\bar{k})}^{(1)}, \mathbf{U}_{(\bar{k})}^{(2)}, \dots, \mathbf{U}_{(\bar{k})}^{(N)}] = \text{parafacALS}(\underline{Y}^{(\bar{k})}, J_t)$

$\mathbf{P}_{(\bar{k})} = (\mathbf{U}_{(\bar{k})}^{(N)\top} \mathbf{A}_{(k_N)}^{(N)}) \oplus \dots \oplus (\mathbf{U}_{(\bar{k})}^{(1)\top} \mathbf{A}_{(k_1)}^{(1)})$ ；

$\mathbf{Q}_{(\bar{k})} = (\mathbf{A}_{(k_N)}^{(N)\top} \mathbf{A}_{(k_N)}^{(N)}) \oplus \dots \oplus (\mathbf{A}_{(k_1)}^{(1)\top} \mathbf{A}_{(k_1)}^{(1)})$ ；

repeat

for $n = 1$ to N do

 for $k_n = 1$ to K_n do

$\mathbf{T} = \mathbf{0}, \mathbf{S} = \mathbf{0}$ ；

 parfor $[\bar{k}]_n = k_n$ do

$\mathbf{P}_{(\bar{k})} = \mathbf{P}_{(\bar{k})} ./ (\mathbf{U}_{(\bar{k})}^{(n)\top} \mathbf{A}_{(k_n)}^{(n)})$

```


$$\mathbf{T} = \mathbf{T} + \mathbf{U}_{(\bar{k})}^{(n)} \mathbf{P}_{(\bar{k})}$$


$$\mathbf{Q}_{(\bar{k})} = \mathbf{Q}_{(\bar{k})} / (\mathbf{A}_{(k_s)}^{(n)\text{T}} \mathbf{A}_{(k_s)}^{(n)})$$


$$\mathbf{S} = \mathbf{S} + \mathbf{Q}_{(\bar{k})}$$


$$\mathbf{A}_{(k_s)}^{(n)} \leftarrow \mathbf{T} \mathbf{S}^{-1}$$

parfor each  $\bar{k}$  do

$$\mathbf{P}_{(\bar{k})} = \mathbf{P}_{(\bar{k})} \oplus (\mathbf{U}_{(\bar{k})}^{(n)\text{T}} \mathbf{A}_{(k_s)}^{(n)})$$


$$\mathbf{Q}_{(\bar{k})} = \mathbf{Q}_{(\bar{k})} \oplus (\mathbf{A}_{(k_s)}^{(n)\text{T}} \mathbf{A}_{(k_s)}^{(n)})$$

until 达到收敛条件;
end

```

3.8 本章小结

本章研究 PARAFAC 模型、 k -秩、可辨识性、PARAFAC 分解、三线性/四线性分解 CRB 分析、自适应 PARAFAC 分解和大规模 PARAFAC 分解。本章内容我们参考了很多文献，如文献[8, 17, 19, 24]，对相关作者表示感谢。对于三线性/四线性模型的应用将在第 6、7 章详细说明。

参考文献

- [1] 刘旭. 基于多维矩阵低秩分解的信号处理技术研究[D]. 南京: 南京航空航天大学, 2009.
- [2] 黄希珠. 极化敏感阵列中基于PARAFAC的盲信源分离算法[D]. 南京: 南京航空航天大学, 2008.
- [3] 张胜男. 基于平行因子分析的阵列参数估计[D]. 南京: 南京航空航天大学, 2008.
- [4] 是莺. 极化敏感阵列中基于PARAFAC的盲DOA和极化估计算法[D]. 南京: 南京航空航天大学, 2009.
- [5] Harshman R A. Foundations of the PARAFAC procedure: models and conditions for an "explanatory" multimodal factor analysis[J]. UCLA Working Papers in Phonetics, 1970, 16: 1-84.
- [6] Kruskal J B. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics[J]. Linear Algebra and its Applications, 1977, 18 (2) : 95-138.
- [7] Sidiropoulos N D, Giannakis G B, Bro R. Blind PARAFAC receivers for DS-CDMA systems[J]. IEEE Transactions on Signal Processing, 2000, 48 (3) : 810-823.

- [8] Sidiropoulos N D, Bro R. On the uniqueness of multilinear decomposition of N-way arrays[J]. Journal of chemometrics, 2000, 14 (3) : 229-239.
- [9] Bro R, Sidiropoulos N D, Giannakis G B. A fast least squares algorithm for separating trilinear mixtures[C]. Proceedings of International Workshop on Independent Component Analysis and Blind Separation, Aussois, France, 1999: 289-294.
- [10] Vorobyov S A, Rong Y, Sidiropoulos N D, et al. Robust iterative fitting of multilinear models[J]. IEEE Transactions on Signal Processing, 2005, 53 (8) : 2678-2689.
- [11] Leshem A, Petrochilos N, van der Veen A. Finite sample identifiability of multiple constant modulus sources [J]. IEEE Transactions on Information Theory, 2003, 49 (9) : 2314-2319.
- [12] Sidiropoulos N D, Bro R, Giannakis G B. Parallel factor analysis in sensor array processing[J]. IEEE Transactions on Signal Processing, 2000, 48 (8) : 2377-2388.
- [13] Jiang T, Sidiropoulos N D, Ten Berge J M. Almost-sure identifiability of multidimensional harmonic retrieval[J]. IEEE Transactions on Signal Processing, 2001, 49 (9) : 1849-1859.
- [14] Li T, Sidiropoulos N D. Blind digital signal separation using successive interference cancellation iterative least squares[J]. IEEE Transactions on Signal Processing, 2000, 48 (11) : 3146-3152.
- [15] Gorman J D, Hero A O. Lower bounds for parametric estimation with constraints[J]. IEEE Transactions on Information Theory, 1990, 36 (6) : 1285-1301.
- [16] Stoica P, Ng B C. On the Cramer-Rao bound under parametric constraints[J]. IEEE Signal Processing Letters, 1998, 5 (7) : 177-179.
- [17] Liu X, Sidiropoulos N D. Cramer-Rao lower bounds for low-rank decomposition of multidimensional arrays[J]. IEEE Transactions on Signal Processing, 2001, 49 (9) : 2074-2086.
- [18] Stoica P, Arye N. MUSIC, maximum likelihood, and Cramer-Rao bound[J]. IEEE Transactions on Acoustics, Speech and Signal Processing, 1989, 37 (5) : 720-741.
- [19] Nion D, Sidiropoulos N D. Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor[J]. IEEE Transactions on Signal Processing, 2009, 57 (6) : 2299-2310.
- [20] De Lathauwer L. A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization[J]. SIAM Journal on Matrix Analysis and Applications, 2006, 28 (3) : 642-666.
- [21] Strobach P. Bi-iteration SVD subspace tracking algorithms[J]. IEEE Transactions on Signal Processing, 1997, 45 (5) : 1222-1240.

- [22] Badeau R, Richard G, David B. Sliding window adaptive SVD algorithms[J]. IEEE Transactions on Signal Processing, 2004, 52 (1) : 1-10.
- [23] Ouyang S, Hua Y. Bi-iterative least-square method for subspace tracking[J]. IEEE Transactions on Signal Processing, 2005, 53 (8) : 2984-2996.
- [24] Huy Phan A, Cichocki A. PARAFAC algorithms for large-scale problems[J]. Neurocomputing, 2011, 74 (11) : 1970-1984.
- [25] Cichocki A, Zdunek R, Phan A H, et al. Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation[M]. Chichester: Wiley. com, 2009.
- [26] Goreinov S A, Tyrtysnikov E E, Zamarashkin N L. A theory of pseudoskeleton approximations[J]. Linear Algebra and its Applications, 1997, 261 (1) : 1-21.

第4章

扩展 PARAFAC 模型

本章将研究扩展 PARAFAC 模型,如基于线性相关平行因子(Parallel Profiles With Linear Dependencies, PARALIND)模型、块状 PARAFAC、PARAFAC2、PARATUCK2 和 TUCKER。

4.1 PARALIND 模型

4.1.1 PARALIND 模型和分解

对于一个三线性模型 $x_{ijk} (i=1, \dots, I, j=1, \dots, J, k=1, \dots, K)$, 其切片形式如下^[1-7]:

$$\mathbf{X}_k = \mathbf{A} \mathbf{H} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T + \mathbf{N}_k, \quad k=1, \dots, K \quad (4.1.1)$$

其中 \mathbf{H} 为相关矩阵。式 (4.1.1) 描述为具有相关结构的 PARAFAC 模型, 即 PARALIND 模型。令 \mathbf{X} 是通过重新排列 $I \times J \times K$ 阵列 $\underline{\mathbf{X}}$ 所得到的 $I \times JK$ 的矩阵:

$$\begin{aligned} \mathbf{X} &= [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K] \\ &= \mathbf{A} \mathbf{H} (\mathbf{C} \odot \mathbf{B})^T + \mathbf{N} \end{aligned} \quad (4.1.2)$$

对于 PARALIND 模型, 求解如下代价函数:

$$\min \left\| \mathbf{X} - \mathbf{A} \mathbf{H} (\mathbf{C} \odot \mathbf{B})^T \right\|_{\text{F}}^2 \quad (4.1.3)$$

阵列的任意正面切片模型为 $\mathbf{X}_k = \mathbf{A} \mathbf{H} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T$, 则

$$\begin{aligned} \text{vec}(\mathbf{X}_k) &= \text{vec}(\mathbf{A} \mathbf{H} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T) \\ &= (\mathbf{B} \mathbf{D}_k(\mathbf{C}) \otimes \mathbf{A}) \text{vec}(\mathbf{H}) \end{aligned} \quad (4.1.4)$$

同时考虑所有的切片, 得到

$$\begin{bmatrix} \text{vec}(X_1) \\ \text{vec}(X_2) \\ \vdots \\ \text{vec}(X_K) \end{bmatrix} = \begin{bmatrix} \mathbf{B}\mathbf{D}_1(\mathbf{C}) \otimes \mathbf{A} \\ \mathbf{B}\mathbf{D}_2(\mathbf{C}) \otimes \mathbf{A} \\ \vdots \\ \mathbf{B}\mathbf{D}_K(\mathbf{C}) \otimes \mathbf{A} \end{bmatrix} \text{vec}(\mathbf{H}) \quad (4.1.5)$$

进而得到

$$\text{vec}(\mathbf{X}^{(I \times JK)}) = [(\mathbf{C} \odot \mathbf{B}) \otimes \mathbf{A}] \text{vec}(\mathbf{H}) \quad (4.1.6)$$

考虑到

$$(\mathbf{C} \odot \mathbf{B})^T (\mathbf{C} \odot \mathbf{B}) = (\mathbf{B}^T \mathbf{B}) \oplus (\mathbf{C}^T \mathbf{C}) \quad (4.1.7)$$

并利用 Kronecker 积和 Khatri-Rao 积的运算法则, 则矩阵 \mathbf{H} 可通过下式更新。

$$\text{vec}(\mathbf{H}) = [(\mathbf{B}^T \mathbf{B}) \oplus (\mathbf{C}^T \mathbf{C}) \otimes (\mathbf{A}^T \mathbf{A})]^{-1} \text{vec} \left(\sum_{k=1}^K \mathbf{A}^T \mathbf{X}_k \mathbf{B} \mathbf{D}_k(\mathbf{C}) \right) \quad (4.1.8)$$

其他承载矩阵也可以通过类似的方法获得。

PARALIND 模型分解过程如下, 初始化所有的承载矩阵后, 算法按如下步骤进行^[1-6]。

$$\text{步骤 1: } \text{vec}(\mathbf{H}) = [(\mathbf{B}^T \mathbf{B}) \oplus (\mathbf{C}^T \mathbf{C}) \otimes (\mathbf{A}^T \mathbf{A})]^{-1} \text{vec} \left(\sum_{k=1}^K \mathbf{A}^T \mathbf{X}_k \mathbf{B} \mathbf{D}_k(\mathbf{C}) \right)$$

$$\text{步骤 2: } \mathbf{A} = \left(\sum_{k=1}^K \mathbf{X}_k \mathbf{B} \mathbf{D}_k(\mathbf{C}) \mathbf{H}^T \right) \left\{ \mathbf{H} [(\mathbf{B}^T \mathbf{B}) \oplus (\mathbf{C}^T \mathbf{C})] \mathbf{H}^T \right\}^{-1}$$

$$\text{步骤 3: } \mathbf{B} = \left(\sum_{k=1}^K \mathbf{X}_k^T \mathbf{A} \mathbf{H} \mathbf{D}_k(\mathbf{C}) \right) \left\{ (\mathbf{H}^T \mathbf{A}^T \mathbf{A} \mathbf{H}) \oplus (\mathbf{C}^T \mathbf{C}) \right\}^{-1}$$

$$\text{步骤 4: } \text{diag}(\mathbf{D}_k(\mathbf{C})) = \left\{ (\mathbf{B}^T \mathbf{B}) \oplus (\mathbf{H}^T \mathbf{A}^T \mathbf{A} \mathbf{H}) \right\}^{-1} \text{diag}(\mathbf{A} \mathbf{H}^T \mathbf{X}_k \mathbf{B}), k=1, \dots, K$$

步骤 5: 跳转到步骤 1 直到收敛。

该算法主要通过推广传统的平行因子交替最小二乘算法而得出, 如果相关矩阵是固定的, 那么更新 \mathbf{H} 的步骤可以跳过。和传统的平行因子的主要区别是, 该算法需要为相关矩阵 \mathbf{H} 建立一个有效的更新^[1-6]。

4.1.2 PARALIND 模型的唯一性

假设

$$\tilde{\mathbf{A}} = [\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \tilde{a}_4 \tilde{a}_5 \tilde{a}_6] = [\mathbf{a}_1 \mathbf{a}_1 \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_2 \mathbf{a}_2] \quad (4.1.9)$$

PARALIND 模型为: $\mathbf{X}_k = \mathbf{A} \mathbf{H} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T$, $k=1, \dots, K$, 其中 $\mathbf{A} = [\mathbf{a}_1 \mathbf{a}_2]$ 。由于 \mathbf{H} 的特殊结构, 该模型可等效为

$$\mathbf{X}_k = \tilde{\mathbf{A}} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T, \quad k=1, \dots, K \quad (4.1.10)$$

考虑另外一个方向的切片形式:

$$\mathbf{X}_i = \mathbf{B} \mathbf{D}_i(\tilde{\mathbf{A}}) \mathbf{C}^T, \quad i=1, \dots, I \quad (4.1.11)$$

由 $\tilde{\mathbf{A}}$ 中的结构可知。考虑如下两个切片:

$$\begin{aligned} X_1 &= BC^T \\ X_2 &= BDC^T \end{aligned} \quad (4.1.12)$$

其中 D 中有多个非零元素, 为对角矩阵, B 和 C 是 F 列的满秩矩阵。令 U 为矩阵 $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$ 的前 F 个左奇异向量, 则 U 张成的空间和 $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$ 张成的空间是相同的, 由于 $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} B \\ BD \end{bmatrix} C^T$ 且 C 满秩, 则 U 张成的空间和 $\begin{bmatrix} B \\ BD \end{bmatrix}$ 张成的空间是相等的, 因此

$$U = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} BT \\ BDT \end{bmatrix} \quad (4.1.13)$$

其中 T 是满秩方阵。定义 $G = T^T B^T B$, 且定义

$$\begin{aligned} R_1 &= U_1^T U_1 = T^T B^T B T = GT \\ R_2 &= U_2^T U_2 = T^T B^T B D T = GDT \end{aligned} \quad (4.1.14)$$

注意到, R_1 和 R_2 满秩且为方阵, 由此得出

$$G^{-1} R_1 = D^{-1} G^{-1} R_2 \quad (4.1.15)$$

或者等效

$$(R_2 R_1^{-1})^T (G^{-1})^T = (G^{-1})^T D \quad (4.1.16)$$

由式 (4.1.16) 可知, D 和 $(G^{-1})^T$ 分别是矩阵 $(R_2 R_1^{-1})^T$ 的特征值和其对应的特征向量, 因此 D 是唯一的。而 $(G^{-1})^T$ 可恢复 (存在一定的模糊), 将 $(G^{-1})^T$ 转置得到 G^{-1} , 由 $R_1 = GT$ 可得到 T 为 $(G^{-1}) R_1$, 由 $U_1 = BT$ 得到 B 为 $U_1 T^{-1}$, 由 $X_1 = BC^T$ 得到 C^T 为 $B^+ X_1$ 。因此 PARALIND 分解为唯一的^[1-6]。

4.2 块状 PARAFAC

4.2.1 块状 PARAFAC 模型

令 $x \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ 是一个三阶张量, 定义两个矩阵集 $\{A^{(q)}\} \in \mathbb{C}^{I_1 \times R_1^{(q)}}$ 和 $\{B^{(q)}\} \in \mathbb{C}^{I_2 \times R_2^{(q)}}$, 其中元素分别为 $a_{i_1, r_1^{(q)}}^{(q)} = [A^{(q)}]_{i_1, r_1^{(q)}}$, $b_{i_2, r_2^{(q)}}^{(q)} = [B^{(q)}]_{i_2, r_2^{(q)}}$, Q 个三阶张量集合, $\{g^{(q)}\} \in \mathbb{C}^{R_1^{(q)} \times R_2^{(q)} \times I_3}$, 其中元素形式为 $c_{r_1^{(q)}, r_2^{(q)}, i_3}^{(q)}$ 。考虑对 x 进行下面的分解^[7,8]:

$$x_{i_1, i_2, i_3} = \sum_{q=1}^Q \sum_{r_1^{(q)}=1}^{R_1^{(q)}} \sum_{r_2^{(q)}=1}^{R_2^{(q)}} a_{i_1, r_1^{(q)}}^{(q)} b_{i_2, r_2^{(q)}}^{(q)} c_{r_1^{(q)}, r_2^{(q)}, i_3}^{(q)} \quad (4.2.1)$$

张量 x_{i_1, i_2, i_3} 被分解为 Q 个三线性块的和形式。这种张量分解构成了一种约束结构的

PARAFAC 分解^[9,10], 它可以看成块张量分解^[11]的一种特殊情况。

定义一个矩阵集 \mathcal{Q} , $\{\mathbf{C}^{(1)}, \mathbf{C}^{(2)}, \dots, \mathbf{C}^{(\mathcal{Q})}\} \in \mathbb{C}^{I_3 \times R_1^{(q)} R_2^{(q)}}$

$$[\mathbf{C}^{(q)}]_{i_3, (r_1^{(q)}-1)R_2^{(q)}+r_2^{(q)}} = c_{r_1^{(q)}, r_2^{(q)}, i_3}^{(q)}, \quad q=1, \dots, \mathcal{Q}$$

其中第 q 个矩阵 $\mathbf{C}^{(q)}$ 与张量 $\mathcal{G}^{(q)}$ 相联系

$$\mathbf{C}^{(q)} = [\text{vec}(\mathbf{C}_1^{(q)\text{T}}), \dots, \text{vec}(\mathbf{C}_{I_3}^{(q)\text{T}})]^{\text{T}}$$

$\mathbf{C}_{i_3}^{(q)\text{T}} \in \mathbb{C}^{R_1^{(q)} \times R_2^{(q)}}$ 是 $\mathcal{G}^{(q)}$ 的第 i_3 个矩阵切片, $i_3=1, \dots, I_3$ 。 χ 的矩阵切片 $\mathbf{X}_{..i_3} \in \mathbb{C}^{I_1 \times I_2}$ 可

以表示为^[7,8]

$$\mathbf{X}_{..i_3} = \sum_{q=1}^{\mathcal{Q}} (\mathbf{A}^{(q)} \otimes \mathbf{1}_{R_2^{(q)}}^{\text{T}}) \mathbf{D}_{i_3}(\mathbf{C}^{(q)}) (\mathbf{1}_{R_1^{(q)}}^{\text{T}} \otimes \mathbf{B}^{(q)})^{\text{T}} \quad (4.2.2)$$

其中, $\mathbf{1}_{R_1^{(q)}}$ 和 $\mathbf{1}_{R_2^{(q)}}$ 为以下标为维度的全 1 向量。

采用下面的等式:

$$\begin{aligned} \mathbf{A}^{(q)} \otimes \mathbf{1}_{R_2^{(q)}}^{\text{T}} &= (\mathbf{A}^{(q)} \otimes \mathbf{I})(\mathbf{I}_{R_1^{(q)}} \otimes \mathbf{1}_{R_2^{(q)}}^{\text{T}}) = \mathbf{A}^{(q)} (\mathbf{I}_{R_1^{(q)}} \otimes \mathbf{1}_{R_2^{(q)}}^{\text{T}}) = \mathbf{A}^{(q)} \boldsymbol{\psi}^{(q)} \\ \mathbf{1}_{R_1^{(q)}}^{\text{T}} \otimes \mathbf{B}^{(q)} &= (\mathbf{1} \otimes \mathbf{B}^{(q)}) (\mathbf{1}_{R_1^{(q)}}^{\text{T}} \otimes \mathbf{I}_{R_2^{(q)}}) = \mathbf{B}^{(q)} (\mathbf{1}_{R_1^{(q)}}^{\text{T}} \otimes \mathbf{I}_{R_2^{(q)}}) = \mathbf{B}^{(q)} \boldsymbol{\Phi}^{(q)} \end{aligned}$$

其中

$$\boldsymbol{\psi}^{(q)} = \mathbf{I}_{R_1^{(q)}} \otimes \mathbf{1}_{R_2^{(q)}}^{\text{T}}, \quad \boldsymbol{\Phi}^{(q)} = \mathbf{1}_{R_1^{(q)}}^{\text{T}} \otimes \mathbf{I}_{R_2^{(q)}} \quad (4.2.3)$$

是约束矩阵, 表示在第 q 个块中的模型交互或不同模式下因子的线性组合, 它们的维度分别为 $R_1^{(q)} \times R_1^{(q)} R_2^{(q)}$ 和 $R_2^{(q)} \times R_1^{(q)} R_2^{(q)}$ 。根据这些定义可以重写式 (4.2.2) 为

$$\mathbf{X}_{..i_3} = \sum_{q=1}^{\mathcal{Q}} \mathbf{A}^{(q)} \boldsymbol{\psi}^{(q)} \mathbf{D}_{i_3}(\mathbf{C}^{(q)}) (\mathbf{B}^{(q)} \boldsymbol{\Phi}^{(q)})^{\text{T}} \quad (4.2.4)$$

定义下面的块矩阵:

$$\begin{aligned} \mathbf{A} &= [\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(\mathcal{Q})}] \in \mathbb{C}^{I_1 \times R_1} \\ \mathbf{B} &= [\mathbf{B}^{(1)}, \dots, \mathbf{B}^{(\mathcal{Q})}] \in \mathbb{C}^{I_2 \times R_2} \\ \mathbf{C} &= [\mathbf{C}^{(1)}, \dots, \mathbf{C}^{(\mathcal{Q})}] \in \mathbb{C}^{I_3 \times R_3} \end{aligned}$$

定义

$$R_1 = \sum_{q=1}^{\mathcal{Q}} R_1^{(q)}, \quad R_2 = \sum_{q=1}^{\mathcal{Q}} R_2^{(q)}, \quad R_3 = \sum_{q=1}^{\mathcal{Q}} R_1^{(q)} R_2^{(q)} \quad (4.2.5)$$

定义块对角约束矩阵为

$$\begin{aligned} \boldsymbol{\Psi} &= \text{BlockDiag}(\boldsymbol{\psi}^{(1)}, \dots, \boldsymbol{\psi}^{(\mathcal{Q})}) \quad (R_1 \times R_3) \\ \boldsymbol{\Phi} &= \text{BlockDiag}(\boldsymbol{\Phi}^{(1)}, \dots, \boldsymbol{\Phi}^{(\mathcal{Q})}) \quad (R_2 \times R_3) \end{aligned} \quad (4.2.6)$$

考虑到这些定义, 式 (4.2.4) 可以表示为简洁的矩阵形式:

$$\mathbf{X}_{..i_3} = \mathbf{A} \boldsymbol{\Psi} \mathbf{D}_{i_3}(\mathbf{C}) (\mathbf{B} \boldsymbol{\Phi})^{\text{T}} \quad (4.2.7)$$

根据下面的对应关系: $\mathbf{A} \rightarrow \mathbf{A} \boldsymbol{\Psi}$, $\mathbf{B} \rightarrow \mathbf{B} \boldsymbol{\Phi}$ 和 $\mathbf{C} \rightarrow \mathbf{C}$ 。因此, 通过仿照式 (4.2.7), $\mathbf{X}_{i=1,2,3}$

可以表示为

$$\mathbf{X}_1 = (\mathbf{C} \odot \mathbf{A} \boldsymbol{\Psi}) (\mathbf{B} \boldsymbol{\Phi})^{\text{T}}, \quad \mathbf{X}_2 = (\mathbf{B} \boldsymbol{\Phi} \odot \mathbf{C})^{\text{T}} (\mathbf{A} \boldsymbol{\Psi})^{\text{T}}, \quad \mathbf{X}_3 = (\mathbf{A} \boldsymbol{\Psi} \odot \mathbf{B} \boldsymbol{\Phi}) \mathbf{C}^{\text{T}} \quad (4.2.8)$$

下面根据前面的等式和定义来讨论这种张量分解^[7-8]:

① 它是三阶张量在 Q 个 PARAFAC 块约束下的张量分解, 它们中的每一个都是以三个矩阵 $\mathbf{A}^{(q)}$ 、 $\mathbf{B}^{(q)}$ 和 $\mathbf{C}^{(q)}$ 为成分的函数。每个成分矩阵即为张量数据在某个方向或模式上的模型。

② 在相同 PARAFAC 块的约束下, 允许不同成分矩阵的列的线性组合合成张量数据。术语交互用于表示这样一种线性组合。

③ 一个块中的交互模式是由约束矩阵 $\boldsymbol{\psi}^{(q)}$ 和 $\boldsymbol{\Phi}^{(q)}$ 来建模的, 可能每个块会不同。

④ 当执行分解计算时, 块间分辨率是 Q 个块可分离率的同义词, 而块内唯一性表示对应块的三个成分矩阵在忽略列模糊和尺度模糊下具有唯一的结果。它依赖于每个块的特殊交互结构。

注意, 式 (4.2.7) 可以解释为在扩张成分矩阵 $\mathbf{A}\boldsymbol{\psi}$ 、 $\mathbf{B}\boldsymbol{\Phi}$ 和 \mathbf{C} 下的约束 PARAFAC 模型^[12,13]。另外, 值得注意的是约束块 PARAFAC 块内结构与 PARALIND 模型^[1-6] 类似。约束块 PARAFAC 模型实际上是一种有块结构的 Trucker2 模型形式, 从某种意义上讲, 它也可以与约束 Trucker3 分解相联系。因此, 约束块 PARAFAC 模型可以与 Trucker2 和 Trucker3^[14] 分析联系起来。块状 PARAFAC 分解如图 4.2.1 所示。

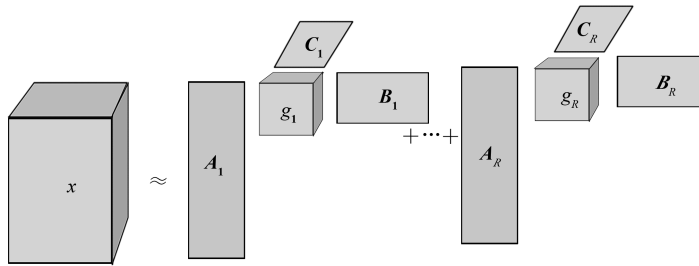


图 4.2.1 块状 PARAFAC 分解^[13]

4.2.2 块状 PARAFAC 分解

定理 4.2.1^[7]: 考虑矩阵切片集合 $\mathbf{X}_{:j_3}$, 假设成分矩阵 $\mathbf{A}^{(q)} \in \mathbb{C}^{I_1 \times R_1^{(q)}}$, $\mathbf{B}^{(q)} \in \mathbb{C}^{I_2 \times R_2^{(q)}}$, $\mathbf{C}^{(q)} \in \mathbb{C}^{I_3 \times R_1^{(q)} R_2^{(q)}}$ 均列满秩。假设每个集合 $\{\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(Q)}\}$, $\{\mathbf{B}^{(1)}, \dots, \mathbf{B}^{(Q)}\}$ 和 $\{\mathbf{C}^{(1)}, \dots, \mathbf{C}^{(Q)}\}$ 都是线性独立的, 如果满足下面的条件:

$$I_1 I_2 \geq R_3, I_1 I_3 \geq R_2, I_2 I_3 \geq R_1 \quad (4.2.9)$$

其中 R_1 、 R_2 、 R_3 由式 (4.2.5) 定义, 可以保证块间唯一性/可分辨性, 同时有非奇异块对角矩阵:

$$\begin{aligned} \mathbf{T}_a &= \text{BlockDiag}(\mathbf{T}_a^{(1)}, \dots, \mathbf{T}_a^{(Q)}) \\ \mathbf{T}_b &= \text{BlockDiag}(\mathbf{T}_b^{(1)}, \dots, \mathbf{T}_b^{(Q)}) \end{aligned} \quad (4.2.10)$$

$$\mathbf{T}_c = \text{BlockDiag}(\mathbf{T}_c^{(1)}, \dots, \mathbf{T}_c^{(Q)})$$

满足

$$(\mathbf{T}_a^{(q)} \otimes \mathbf{T}_b^{(q)})^{-1} = \mathbf{T}_c^{(q)\top}, \quad q=1, \dots, Q \quad (4.2.11)$$

在不考虑列模糊的情况下，这样的 $\bar{\mathbf{A}} = \mathbf{A}\mathbf{T}_a$ 、 $\bar{\mathbf{B}} = \mathbf{B}\mathbf{T}_b$ 和 $\bar{\mathbf{C}} = \mathbf{C}\mathbf{T}_c$ 构成相同的矩阵组 $\{\mathbf{X}\}_{i=1,2,3}$ 。

定义 $\bar{R}_1 = \max(R_1^{(1)}, \dots, R_1^{(Q)})$ 和 $\bar{R}_2 = \max(R_2^{(1)}, \dots, R_2^{(Q)})$ ，式 (4.2.9) 的条件可以重写为以下更简单的形式：

$$I_1 I_2 \geq Q \bar{R}_1 \bar{R}_2, \quad I_1 I_3 \geq Q \bar{R}_2, \quad I_2 I_3 \geq Q \bar{R}_1$$

等价于

$$\min\left(\left\lfloor \frac{I_1 I_2}{\bar{R}_1 \bar{R}_2} \right\rfloor, \left\lfloor \frac{I_1 I_3}{\bar{R}_2} \right\rfloor, \left\lfloor \frac{I_2 I_3}{\bar{R}_1} \right\rfloor\right) \geq Q \quad (4.2.12)$$

其中 $\lfloor \bullet \rfloor$ 表示向下取整。式 (4.2.12) 可以保证块 PARAFAC 的块间唯一性。

块对角矩阵 \mathbf{T}_a 、 \mathbf{T}_b 和 \mathbf{T}_c 意味着旋转自由度被限制在块内，换句话说，约束块 PARAFAC 模型有块间唯一性。然而，通常在 $R_1^{(q)} \geq 2$ 和 $R_2^{(q)} \geq 2$ 时，块内存在非唯一性。例如，如果 \mathbf{C} 已知，块间唯一性可以恢复。否则，如果 \mathbf{C} 有一些特殊结构（如对角矩阵、Toeplitz 矩阵等），为了限制块内非唯一性，在模型计算中可以增强这些潜在结构。

4.3 PARAFAC2

4.3.1 PARAFAC2 模型

Harshman 在 1972 年提出了 PARAFAC2 模型^[16]。严格意义上说，PARAFAC2 模型不是一个独立的张量分解模型，它可以看成 PARAFAC 模型的一种变形。这里给出 PARAFAC2 模型的表达式，定义矩阵集合 \mathbf{X}_k ， $k=1, \dots, K$ ，其中， \mathbf{X}_k 是 $I_k \times J$ 的矩阵， I_k 的值随着 k 的改变而改变，假设 PARAFAC2 分解的维度为 R ，此时，PARAFAC2 具有以下形式^[15]：

$$\mathbf{X}_k \approx \mathbf{U}_k \mathbf{S}_k \mathbf{V}^\top \quad (4.3.1)$$

其中， \mathbf{U}_k 为 $I_k \times R$ 的矩阵， \mathbf{S}_k 为 $R \times R$ 的对角矩阵， $k=1, \dots, K$ 。 \mathbf{V} 为 $J \times R$ 的因子矩阵，注意，在式 (4.3.1) 中，索引 k 取不同的值时矩阵 \mathbf{V} 保持不变。图 4.3.1 给出了 PARAFAC2 分解的示意图。

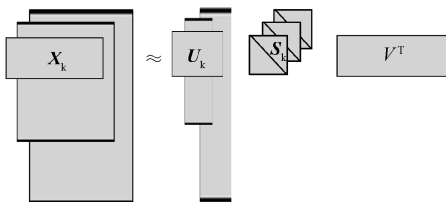


图 4.3.1 PARAFAC2 分解^[13]

从式(4.3.1)可以看出, PARAFAC2 模型去除了 PARAFAC 模型中的一些约束条件。例如, 在 PARAFAC 模型中, 所有矩阵集合具有相同的因子, 然而, 在 PARAFAC2 模型中, 仅有一个模式矩阵集合具有相同的因子, 而其他的模式矩阵集合可以具有不同的因子。因此, 与 PARAFAC 模型相比, PARAFAC2 模型的应用将更加广泛, 该模型不仅可以在较少的约束条件下对常规三线性张量进行近似处理, 而且, 当数据模型中的数据矩阵的维度在某一个模式上发生变化时(如矩阵列维相同、行维不同的情况), PARAFAC2 模型依然有效。

这里需要指出的是, 在没有额外的约束条件时, PARAFAC2 分解是不唯一的。例如, 定义 \mathbf{T} 为 $R \times R$ 的非奇异矩阵, $\mathbf{F}_k, k=1, \dots, K$ 为 K 个 $R \times R$ 的对角矩阵, 有

$$\mathbf{U}_k \mathbf{S}_k \mathbf{V}^T = (\mathbf{U}_k \mathbf{S}_k \mathbf{T}^{-1} \mathbf{F}^{-1}) \mathbf{F}_k (\mathbf{V} \mathbf{T}^T)^T = \mathbf{G}_k \mathbf{F}_k \mathbf{W}^T \quad (4.3.2)$$

式(4.3.2)表明, PARAFAC2 分解是不唯一的。为了改进 PARAFAC2 模型的唯一性, Harshman 在 PARAFAC2 模型中引入了约束条件, 对式(4.3.1)中的 \mathbf{U}_k 进行约束, 即 $\Phi = \mathbf{U}_k^T \mathbf{U}_k, k=1, \dots, K$ 。此时, PARAFAC2 模型可以表示为

$$\mathbf{X}_k \approx \mathbf{Q}_k \mathbf{H} \mathbf{S}_k \mathbf{V}^T \quad (4.3.3)$$

其中, $\mathbf{U}_k = \mathbf{Q}_k \mathbf{H}$, \mathbf{Q}_k 为 $I_k \times R$ 的正交矩阵, \mathbf{H} 是 $R \times R$ 的方阵, 有

$$\mathbf{U}_k^T \mathbf{U}_k = \mathbf{H}^T \mathbf{Q}_k^T \mathbf{Q}_k \mathbf{H} = \mathbf{H}^T \mathbf{H} = \Phi \quad (4.3.4)$$

4.3.2 PARAFAC2 分解

研究表明, 可以通过两种方式对 PARAFAC2 分解进行拟合, 一种方式是对相关矩阵的叉乘进行拟合(间接拟合^[17]), 另一种方式是对式(4.3.1)中的原始数据进行拟合(直接拟合^[18])。间接拟合方法通过对叉乘 $\mathbf{X}_k^T \mathbf{X}_k \approx \mathbf{V} \mathbf{S}_k \Phi \mathbf{S}_k \mathbf{V}^T, k=1, \dots, K$ 进行拟合处理, 得到 $\mathbf{V}, \mathbf{S}_k, \Phi$ 。通过对 Φ 进行半正定约束, 间接拟合可以使用 DEDICOM 分解算法实现。直接拟合方法可以通过文献[18]给出的迭代方法实现。在该方法中, 每次迭代分为两步, 首先利用 SVD 求解 \mathbf{Q}_k , 然后利用 PARAFAC-ALS 方法更新 $\mathbf{H}, \mathbf{S}_k, \mathbf{V}$ 。

PARAFAC2 模型的分解唯一性与多种因素有关, 如矩阵的个数(K)、 Φ 的正定性、 \mathbf{A} 的满秩性和 \mathbf{S}_k 的非奇异性等^[19, 20], 一定条件下 PARAFAC2 模型可以实现分解唯一。

4.4 PARATUCK2

4.4.1 PARATUCK2 模型

Harshman 和 Lundy 提出了 PARATUCK2 分解模型^[19]。由于 PARATUCK2 分解可以看成 PARAFAC 分解和 TUCKER2 分解的结合, 因此被命名为 PARATUCK2 分解。

PARATUCK2 模型是 DEDICOM 的扩展模型，这里给出该模型的表达式。考虑三维张量 $\underline{X} \in \mathbb{R}^{I \times J \times K}$ ，PARATUCK2 分解的目标是将模式 1 中的目标值分组为 P 个潜元 (Latent Components)，将模式 2 中的目标值分组为 Q 个潜元。式 (4.4.1) 给出了 PARATUCK2 分解的形式^[15]：

$$\underline{X} \approx \underline{A} \underline{D}_k^A \underline{R} \underline{D}_k^B \underline{B}^T, \quad k=1, \dots, K \quad (4.4.1)$$

其中， $\underline{A} \in \mathbb{R}^{I \times P}$ 、 $\underline{B} \in \mathbb{R}^{J \times Q}$ 、 $\underline{R} \in \mathbb{R}^{P \times Q}$ ，同时， $\underline{D}_k^A \in \mathbb{R}^{P \times P}$ 、 $\underline{D}_k^B \in \mathbb{R}^{Q \times Q}$ 是对角矩阵。矩阵 \underline{A} 和 \underline{B} 的列与潜因子 (Latent Factor) 有关，因此， b_{jq} 中的变量 j 是指第 j 个目标值，而变量 q 是指第 q 个潜元。同样，对角矩阵 \underline{D}_k 的元素值表示每个潜元相对于第三维度的相关性。矩阵 \underline{R} 表示 \underline{A} 中的 P 个潜元和 \underline{B} 中的 Q 个潜元之间的相关性。对角矩阵 $\underline{D}_k^A, \underline{D}_k^B$ 可以变换为张量 \underline{D}^A 和 \underline{D}^B 的形式。根据上面的描述，PARATUCK2 模型分解如图 4.4.1 所示。

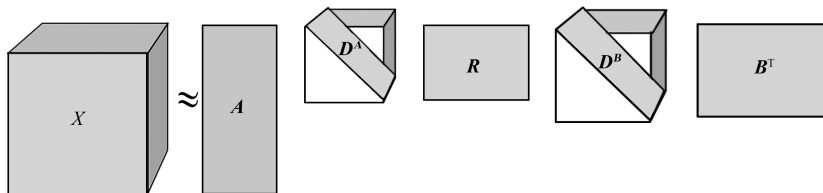


图 4.4.1 PARATUCK2 模型^[13]

4.4.2 PARATUCK2 分解

Harshman 和 Lundy 对 PARATUCK2 模型的唯一性进行了讨论。在 $P=Q$ 且 \underline{R} 矩阵元素非零的条件下，文献[19]证明了传统 PARATUCK2 模型和对称加权 PARATUCK2 模型 ($\underline{D}_k^A = \underline{D}_k^B$) 的轴方向的唯一性。Bro 对 PARATUCK2 模型的参数计算进行了讨论，给出了适用于 PARATUCK2 模型的迭代最小二乘算法^[21]。该迭代算法与 Kiers 的 DEDICOM 算法类似^[18]。

4.5 TUCKER

4.5.1 TUCKER 模型

Tucker 于 1963 年第一次给出了 TUCKER 分解模型^[22]，随后，Levin 和 Tucker 重新定义了该分解模型^[23-25]。和 PARAFAC 模型类似，Tucker 分解有着多种名称，表 4.5.1 总结了 Tucker 分解几种常见的名称。

表 4.5.1 TUCKER 分解名称简介^[15]

名称	提出者
Three-mode factor analysis (3MFA/Tucker3)	Tucker, 1966 年 ^[25]
Three-mode principal component analysis (3MPCA)	Kroonenberg 和 De Leeuw, 1980 年 ^[26]
N-mode principal components analysis	Kapteyn 等, 1986 年 ^[27]
Higher-order SVD (HOSVD)	De Lathauwer 等, 2000 年 ^[26]
N-mode SVD	Vasilescu 和 Terzopoulos, 2002 年 ^[27]

TUCKER 分解是高维独立主元分析的一种形式, 它将张量分解为一个核张量与模式矩阵相乘的形式。下面给出 TUCKER 模型分解的表达式。定义三维张量 $\chi \in \mathbb{R}^{I \times J \times K}$, TUCKER 分解具有以下形式:

$$\chi = \mathcal{T} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{pqr} \mathbf{a}_p \circ \mathbf{b}_q \circ \mathbf{c}_r = [\mathcal{T}; \mathbf{A}, \mathbf{B}, \mathbf{C}] \quad (4.5.1)$$

其中, $\mathbf{A} \in \mathbb{R}^{I \times P}$ 、 $\mathbf{B} \in \mathbb{R}^{J \times Q}$ 、 $\mathbf{C} \in \mathbb{R}^{K \times R}$ 是三个因子矩阵 (通常情况下, 三个因子矩阵彼此正交), 同时, 三个因子矩阵可以看成每个模式中的独立主元。张量 $\mathcal{T} \in \mathbb{R}^{P \times Q \times R}$ 是核矩阵, 该矩阵中的值表示不同成分间的相互作用程度。TUCKER 分解的标量表达式为

$$x_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R g_{pqr} a_{ip} b_{jq} c_{kr}, i=1, \dots, I, j=1, \dots, J, k=1, \dots, K \quad (4.5.2)$$

其中, P, Q, R 是因子矩阵 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 中成分的数目。如果 P, Q, R 的值小于 I, J, K , 则核矩阵 \mathcal{T} 可以看成 χ 的压缩形式。在一些情况下, 三维张量压缩后的数据量要比原始数据小很多, 易于存储。图 4.5.1 为 TUCKER 模型的示意。

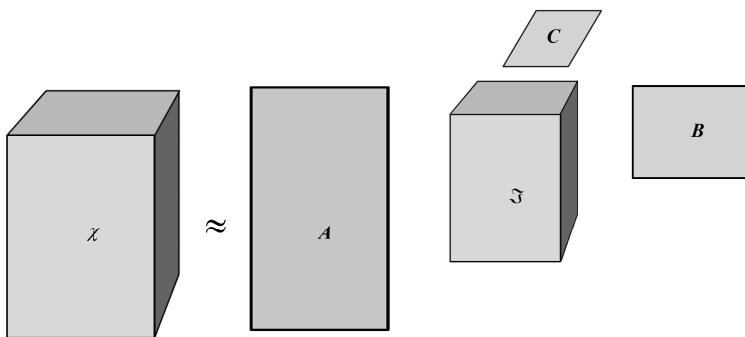


图 4.5.1 TUCKER 分解示意^[13]

TUCKER 分解在每一个模式下的矩阵形式如下:

$$\begin{aligned} X_{(1)} &\approx \mathbf{A} \mathbf{G}_{(1)} (\mathbf{C} \otimes \mathbf{B})^T \\ X_{(2)} &\approx \mathbf{B} \mathbf{G}_{(2)} (\mathbf{C} \otimes \mathbf{A})^T \\ X_{(3)} &\approx \mathbf{C} \mathbf{G}_{(3)} (\mathbf{B} \otimes \mathbf{A})^T \end{aligned} \quad (4.5.3)$$

其中, $\mathbf{G}_{(1)}, \mathbf{G}_{(2)}, \mathbf{G}_{(3)}$ 为核矩阵的三个切面形式。同样, TUCKER 模型可以扩展到 N 维张量的形式:

$$\chi = \mathcal{T} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 \cdots \times_N \mathbf{A}^{(N)} = \llbracket \mathcal{T}; \mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \dots, \mathbf{A}^{(N)} \rrbracket$$

其元素形式和模式矩阵形式分别为

$$x_{i_1 i_2 \dots i_N} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \cdots \sum_{r_N=1}^{R_N} g_{r_1 r_2 \dots r_N} a_{i_1 r_1}^{(1)} a_{i_2 r_2}^{(2)} \dots a_{i_N r_N}^{(N)}, i_n = 1, \dots, I_n, n = 1, \dots, N \quad (4.5.4)$$

$$\mathbf{X}_{(n)} = \mathbf{A}^{(n)} \mathbf{G}_{(n)} (\mathbf{A}^{(N)} \otimes \dots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \dots \otimes \mathbf{A}^{(1)})^T \quad (4.5.5)$$

4.5.2 TUCKER 分解

1966 年, Tucker 给出了三种计算 TUCKER 分解的方法, 但是, 由于当时计算水平的限制, 要计算一个大维度矩阵的奇异值分解非常困难。De Lathauwer、De Moor 和 Vandewalls 对该方法进行了研究, 将其命名为高阶奇异值分解 (High-Order SVD, HOSVD), 该方法是矩阵 SVD 的一种扩展^[27]。目前, TUCKER 分解已在化学分析、心理测试学、信号处理和计算机视觉等众多领域中得到了广泛的应用。

4.6 本章小结

本章研究了扩展 PARAFAC 模型, 如 PARALIND 模型、块状 PARAFAC、PARAFAC2、PARATUCK2 和 TUCKER。本章我们参考了很多文献, 如文献[1,7,13], 对相关作者表示感谢。对于 PARALIND 模型的应用将在第 8 章详细说明。

参考文献

- [1] Bro R, Harshman R A, Sidiropoulos N D, et al. Modeling multi-way data with linearly dependent loadings[J]. Journal of Chemometrics, 2009, 23 (7-8) : 324-340.
- [2] Bahram M, Bro R. A novel strategy for solving matrix effect in three-way data using parallel profiles with linear dependencies[J]. Analytica chimica acta, 2007, 584 (2) : 397-402.
- [3] Zhang X, Feng G, Xu D, et al. blind PARALIND space-time multiuser detection for asynchronous CDMA SYSTEM[J]. Journal of Circuits, Systems, and Computers, 2009, 18 (3) : 503-517.

- [4] Zhang X, Gao X, Wang Z. Blind paralind multiuser detection for smart antenna CDMA system over multipath fading channel [J]. Progress In Electromagnetics Research, 2009, 89: 23-38.
- [5] Xiaofei Z, Fei W, Dazhuan X. Blind signal detection algorithm for MIMO-OFDM systems over multipath channel using PARALIND model [J]. IET Communications, 2011, 5 (5) : 606-611.
- [6] Zhang X, Zhou M, Li J. A PARALIND Decomposition-Based Coherent Two-Dimensional Direction of Arrival Estimation Algorithm for Acoustic Vector-Sensor Arrays [J]. Sensors, 2013, 13 (4) : 5302-5316.
- [7] Cavalcante C C, Colares R F, Barbosa P C. Telecommunications: Advances and Trends in Transmission, Networking and Applications[M]. University of Fortaleza Press, 2006: 57-82.
- [8] De Almeida A L, Favier G, Mota J C M. PARAFAC-based unified tensor modeling for wireless communication systems with application to blind multiuser equalization[J]. Signal Processing, 2007, 87 (2) : 337-351.
- [9] De Lathauwer L. The decomposition of a tensor in a sum of rank- (R_1, R_2, R_3) terms[C]. Workshop on Tensor Decompositions and Applications, 2005.
- [10] Harshman R A. Foundations of the PARAFAC procedure: models and conditions for an “explanatory” multimodal factor analysis[J]. UCLA Working Papers in Phonetics, 1970, 16: 1-84.
- [11] Carroll J D, Chang J. Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition[J]. Psychometrika, 1970, 35 (3) : 283-319.
- [12] Bro R. Multi-way analysis in the food industry: models, algorithms, and applications[D]. Amsterdam: University of Amsterdam, 1998.
- [13] Kolda T G, Bader B W. Tensor decompositions and applications[J]. SIAM review, 2009, 51 (3) : 455-500.
- [14] Harshman R A. PARAFAC2: Mathematical and technical notes[J]. UCLA working papers in phonetics, 1972, 22: 30-44.
- [15] Kiers H A. An alternating least squares algorithm for PARAFAC2 and three-way DEDICOM[J]. Computational Statistics & Data Analysis, 1993, 16 (1) : 103-118.
- [16] Kiers H A, Ten Berge J M, Bro R. PARAFAC2-Part I. A direct fitting algorithm for the PARAFAC2 model[J]. Journal of Chemometrics, 1999, 13 (3-4) : 275-294.
- [17] Harshman R A, Lundy M E. Uniqueness proof for a family of models sharing features of Tucker's three-mode factor analysis and PARAFAC/CANDECOMP[J]. Psychometrika, 1996, 61 (1) : 133-154.

- [18] Ten Berge J M, Kiers H A. Some uniqueness results for PARAFAC2[J]. Psychometrika, 1996, 61 (1) : 123-132.
- [19] Smilde A, Bro R, Geladi P. Multi-way analysis: applications in the chemical sciences[M]. Wiley. com, 2005.
- [20] Tucker L R. Implications of factor analysis of three-way matrices for measurement of change[J]. Problems in measuring change, 1963: 122-137.
- [21] Levin J. Three-mode factor analysis[J]. Psychological Bulletin, 1965, 64 (6) : 442.
- [22] Tucker L R. The extension of factor analysis to three-dimensional matrices[J]. Contributions to mathematical psychology, 1964: 109-127.
- [23] Tucker L R. Some mathematical notes on three-mode factor analysis[J]. Psychometrika, 1966, 31 (3) : 279-311.
- [24] Kroonenberg P M, De Leeuw J. Principal component analysis of three-mode data by means of alternating least squares algorithms[J]. Psychometrika, 1980, 45 (1) : 69-97.
- [25] Kapteyn A, Neudecker H, Wansbeek T. An approach ton-mode components analysis[J]. Psychometrika, 1986, 51 (2) : 269-275.
- [26] De Lathauwer L, De Moor B, Vandewalle J. A multilinear singular value decomposition[J]. SIAM journal on Matrix Analysis and Applications, 2000, 21 (4) : 1253-1278.
- [27] Vasilescu M A O, Terzopoulos D. Multilinear analysis of image ensembles: Tensorfaces[M]. Springer, 2002: 447-460.

第5章

PARAFAC 压缩感知模型

本章将研究压缩感知基本原理，以及平行因子压缩感知原理。

5.1 压缩感知基本原理

压缩感知 (Compressive Sensing, CS) 理论是 2006 年由 Candes、Romberg、Tao 和 Donoho 等人提出的一套关于稀疏信号采集和恢复的新理论^[1-4]。CS 理论充分利用信号稀疏性或可压缩性，在信号采样的同时，还对数据进行适当的压缩，大大减轻了数据传输、存储、处理的负担，与传统的奈奎斯特采样理论相比，CS 理论采样速率不取决于信号的带宽，而决定于信息在信号中的结构和内容。因此，该理论一经提出就成为信息论^[5,6]、信号/图像处理^[7]、无线通信^[8]，以及超宽带系统中的信号检测^[9,10]、信道估计^[11,12]和参数估计^[13,14]等众多领域的研究热点。

信号的稀疏性是压缩感知的重要前提和理论基础，信号的稀疏性定义如下。

定义 5.1.1: 信号的稀疏性为信号中非零元素数目较少。

定义 5.1.2: 信号在某个变换域下近似稀疏，即可压缩信号。或者从理论上讲，任何信号都具有可压缩性，只要能找到其相应的稀疏表示空间，就可以有效地进行压缩采样。

定义 5.1.3: 矩阵奇异值的稀疏性，即矩阵奇异值中非零元素的个数（矩阵的秩）相对较少；也称矩阵的低秩性，即矩阵的秩相对于矩阵的行数或列数而言很小。

5.1.1 压缩感知的理论框架

本节将详细介绍压缩感知理论的具体三个核心的部分：一是信号的稀疏表示，二是测量矩阵的设计，三是信号重构算法。

1. 信号的稀疏表示

为了更清晰地描述信号的稀疏表示问题，首先定义矢量 $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$ 的 ℓ_p 范数：

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^N |x_i|^p \right)^{1/p}$$

对于信号 $\mathbf{x} \in R^N$ ，其在标准正交基 Ψ 下的表示系数矢量为

$$\mathbf{a} = \Psi^T \mathbf{x}$$

根据 ℓ_p 范数的定义，若 \mathbf{a} 满足

$$\|\mathbf{a}\|_p \leq K$$

对于实数 $0 < p < 2$ 和 $K > 0$ 同时成立，则称 \mathbf{x} 在变换域 Ψ 下是稀疏的。特别地，当 $p=0$ 时，称信号 \mathbf{x} 在变换域 Ψ 下是 K 稀疏的。

首先考虑一般的信号重构问题，信号 \mathbf{x} 在时域本身就是稀疏或可压缩的，即上述的基变换 Ψ 为 Dirac 函数。给定一个投影测量矩阵 $\Phi \in R^{M \times N}$ ($M \ll N$)，则信号 \mathbf{x} 在该测量矩阵 Φ 下的线性投影测量值为^[15]

$$\mathbf{y} = \Phi \mathbf{x} \quad (5.1.1)$$

现在考虑由投影测量数据 \mathbf{y} 来重构信号 \mathbf{x} 。由于 \mathbf{y} 的维数 M 远小于 \mathbf{x} 的维数 N ，方程 (5.1.1) 是欠定方程，有无穷多个解，直接通过解方程的方法无法重构原始信号。然而，理论已经证明，如果原始信号 \mathbf{x} 本身在时域是 K 稀疏的或可压缩的，并且 \mathbf{y} 与 Φ 满足一定的条件，信号 \mathbf{x} 可以由投影测量值 \mathbf{y} 通过求解以下最小 ℓ_0 范数问题以极高概率得到其精确的重构。

常见的自然信号在时域内几乎都是不稀疏的。然而，从傅里叶变换到小波变换和多尺度几何分析，为解决上面提出的问题提供了思路，即寻找待处理信号在某变换基域更为稀疏的表示方式。设自然信号 \mathbf{x} 在基变换 Ψ 下具有稀疏性或可压缩性，即 $\mathbf{x} = \Psi \mathbf{a}$ ， \mathbf{a} 为信号 \mathbf{x} 在变换域 Ψ 下的 K 稀疏表示系数。于是，信号 \mathbf{x} 在投影测量矩阵 Φ 下线性测量可以表示为^[15]

$$\mathbf{y} = \Phi \mathbf{x} = \Psi \mathbf{a} = \tilde{\Phi} \mathbf{a} \quad (5.1.2)$$

其中， $\tilde{\Phi} = \Phi \Psi$ 为 $M \times N$ 维矩阵，表示推广后的测量矩阵，称为感知矩阵 (Sensing Matrix)，那么， \mathbf{y} 可以看成稀疏信号 \mathbf{a} 关于测量矩阵 $\tilde{\Phi}$ 的线性测量。由于变换域 Ψ 是固定的，要使 $\tilde{\Phi} = \Phi \Psi$ 满足 RIP 条件，则投影测量矩阵 $\tilde{\Phi}$ 就必须满足一定的条件。

2. 测量矩阵的设计

在压缩感知理论中, 得到信号的稀疏表示以后, 设计一个测量矩阵 Φ , 使得在该测量矩阵上的压缩投影得到的 M 个测量值能够保留原始信号的绝大部分信息, 使原始信号的信息损失最小, 从而保证从这些少量的测量值中能够精确重构出长度为 N ($M \ll N$) 的原始信号。

测量矩阵设计是压缩采样理论的核心, 直接决定了压缩采样理论是否能够成功实现。由于压缩测量个数和信号重建精度以及信号稀疏性有着密切的联系, 因此测量矩阵的设计应该与稀疏字典的设计统筹考虑。从原理的角度看, 测量矩阵的设计要以非相干性或等距约束性为基本准则, 既要减少压缩测量个数又要确保压缩感知的信号重建精度。从技术角度看, 测量矩阵的设计包括两方面: 一是测量矩阵的元素, Candès 等人给出了随机生成的设计策略; 二是测量矩阵的维数, 压缩测量个数 M 与信号稀疏性 K 和信号长度 N 应该满足一定的关系。

压缩感知理论框架下的测量数据 $y = \Phi x$, 其中测量矩阵 Φ 的维数为 $M \times N$, 且有 $M \ll N$ 。常用测量矩阵有以下几类[15,16]。

① 高斯随机矩阵。对于一个 $M \times N$ 维高斯随机矩阵, 当 $M \geq CK \log(N/K)$ 时, 感知矩阵 Φ 在较大概率下具有 RIP 性质。

② 二值随机矩阵。二值随机矩阵是指矩阵中每个值都服从对称伯努利分布。伯努利分布的 ± 1 矩阵便于硬件系统实现。

③ 局部傅里叶矩阵。局部傅里叶矩阵是从傅里叶矩阵中随机抽取 M 行, 再对其进行单位正则化得到的矩阵。局部傅里叶矩阵的一个很好优点是可以利用 FFT 矩阵得到, 降低了采样系统的复杂性。

④ 其他测量矩阵。包括局部 Hadamard 矩阵、Toeplitz 矩阵和循环矩阵等。

3. 信号重构算法

信号重构是压缩感知理论的关键部分, 其目的是从 M 个测量值中重构出长度为 N ($M \ll N$) 的稀疏信号。从表面上看, 这是一个无法直接求解的欠定方程, 由于信号是稀疏的或可压缩的, 若感知矩阵满足 RIP 等稀疏重构条件, 则该信号是可以以很高的概率被稀疏重构出来的。重建算法的设计应该遵循如下基本准则: 算法应该利用尽可能少的压缩测量快速、稳定、精确或近似精确地重建原始信号。

E. Candès 等证明, 信号重构问题可以通过求解以下最小 ℓ_0 范数问题求解^[15]:

$$\hat{a} = \arg \min \|a\|_0 \text{ s.t. } \Phi a = y \quad (5.1.3)$$

显然, 上述问题求解需要列出 a 中所有非零项位置的 $\binom{N}{K}$ 种可能的组合才能得到最优解。因此, ℓ_0 范数最小化是个需要组合搜索的 NP-hard 问题。当 N 很大时, 不仅在数值计算上无法有效实现, 而且抗噪能力很差。为此, 学者们陆续提出了多种近似等价的信号重建算法。简单地说, 主要包括三类方法: 松弛方法、贪婪方法、非凸方法。

最小 ℓ_0 范数问题是一个 NP 难题，而文献[17]指出，采用 ℓ_1 范数代替 ℓ_0 范数可以凸优化来求解：

$$\hat{a} = \arg \min \|a\|_1 \quad \text{s.t. } \tilde{\Phi}a = y \quad (5.1.4)$$

文献[18]证明了在满足一定条件时问题 (5.1.3) 和问题 (5.1.4) 是等价的，因此，信号重构问题可以转化为一个线性规划问题加以求解，这种方法也称基追踪 (Basis Pursuit, BP) 方法。如果考虑噪声的情况，上述问题可以转化为以下的最小 ℓ_1 范数问题：

$$\hat{a} = \arg \min \|a\|_1 \quad \text{s.t. } \|\tilde{\Phi}a - y\| \leq \sigma \quad (5.1.5)$$

其中， σ 代表噪声一个可能的标准差。关于最小 ℓ_1 范数问题一些文献相继提出内点法、最小角回归、梯度投影、软/硬迭代阈值等多种稀疏重建算法。总的来说，此类方法重建精度高，但需要的压缩测量个数少。

5.1.2 矩阵秩最小化理论

信号的稀疏性另外一个定义为：矩阵奇异值的稀疏性，即矩阵奇异值中非零元素的个数（矩阵的秩）相对较少；也称矩阵的低秩性，即矩阵的秩相对于矩阵的行数或列数而言很小。

与压缩传感紧密相关的一个问题是矩阵秩最小化问题。矩阵秩最小化就是利用矩阵奇异值的稀疏性。低秩矩阵模型在通信信号处理等领域具有一定的应用，这往往涉及仿射矩阵秩最小化 (Affine Rank Minimization) 的问题^[19,20]：

$$\begin{aligned} \min_X \quad & \text{rank}(X) \\ \text{s.t.} \quad & A(X) = b \end{aligned} \quad (5.1.6)$$

其中， $X \in \mathbf{R}^{m \times n}$ 是决策变量，真实的决策变量 X_0 具有低秩特性， A 是线性映射， $A: \mathbf{R}^{m \times n} \rightarrow \mathbf{R}^p$ ，将决策变量 X 映射到观测变量 $b \in \mathbf{R}^p$ 。目标函数是数据 X 矩阵的秩，即其奇异值构成向量的稀疏性。然而，通过上述问题的求解是 NP 难的。注意到，函数 $\text{rank}(X)$ 在集合 $\{X \in \mathbf{R}^{m \times n} : \|X\| \leq 1\}$ 上的凸包 (Convex Envelop) 是 X 的核范数 $\|X\|_* = \sum_{k=1}^n \sigma_k(X)$ (矩阵 X 的所有奇异值之和)。研究人员转而求解以下的凸优化问题^[19,20]：

$$\begin{aligned} \min_X \quad & \|X\|_* \\ \text{s.t.} \quad & A(X) = b \end{aligned} \quad (5.1.7)$$

矩阵秩最小化的一个典型应用是低秩矩阵填充 (Low-rank Matrix Completion) 问题^[19]。假定原始数据矩阵是低秩的，但是矩阵中含有很多未知的元素。从一个不完整的矩阵中恢复出一个完整的低秩矩阵，便是低秩矩阵填充问题。例如，著名的 Netflix 问题便是一个典型的低秩矩阵填充问题。Netflix 是一家在线影片租赁提供商，公司能够提供超大数量的 DVD，而且能够让顾客快速方便地挑选影片，同时免费递送。Netflix 大奖赛从 2006 年 10 月

开始, Netflix 公开了大约 1 亿个 1~5 的匿名影片评级, 数据集仅包含了影片名称、评价星级和评级日期, 没有任何文本评价的内容。比赛要求参赛者预测 Netflix 的客户分别喜欢什么影片, 要把预测的效率提高 10% 以上。这个问题可以用矩阵填充来进行建模。假设矩阵的每行代表同一用户对不同电影的打分, 每列代表不同用户对同一电影的打分。用户数量巨大, 电影数目巨大, 因此这个矩阵的维度十分大。由于用户所打分的电影有限, 这个矩阵中只有很小一部分的元素值已知, 而且可能含有噪声或误差。

数学上, 从观测到的不完整的矩阵 $\mathbf{M} \in \mathbf{R}^{m \times n}$, Ω 对应于观测到的 \mathbf{M} 中元素对应的位置集合, 即若 \mathbf{M}_{ij} 被观测到, 则 $(i, j) \in \Omega$ 恢复出完整的低秩矩阵, 秩为 r , 即^[20-21]:

$$\begin{aligned} \min_X \quad & \text{rank}(\mathbf{X}) \\ \text{s.t.} \quad & \mathbf{X}_{ij} = \mathbf{M}_{ij}, (i, j) \in \Omega \end{aligned} \quad (5.1.8)$$

但是, 对上述优化问题的求解是 NP 难题, 且问题求解的复杂度还随着矩阵维数的增长呈平方倍指数增长。取而代之, 一般采用以下的凸优化问题^[21]:

$$\begin{aligned} \min_X \quad & \|\mathbf{X}\|_* \\ \text{s.t.} \quad & \mathbf{X}_{ij} = \mathbf{M}_{ij}, (i, j) \in \Omega \end{aligned} \quad (5.1.9)$$

5.2 PARAFAC 压缩感知理论

文献[22]将压缩感知理论推广到了高阶张量信号处理, 提出了平行因子压缩感知理论的框架。

5.2.1 张量分解的基础

TUCKER3 模型和 PARAFAC 模型是两种基本的张量模型。TUCKER3 通常是不可辨识的, 但在数据压缩方面很有用处。PARAFAC 在满足特定条件时是可辨识的。

TUCKER3: 考虑一个由 K 个矩阵片 $\{\mathbf{X}_k\}_{k=1}^K$ 组成的 $I \times J \times K$ 的三维阵列 $\underline{\mathbf{X}}$, 将 $\{\mathbf{X}_k\}_{k=1}^K$ 组织成高矩阵 $\mathbf{X} = [\text{vec}(\mathbf{X}_1), \dots, \text{vec}(\mathbf{X}_K)]$ 的形式。TUCKER3 模型可以被写成 $\mathbf{X} \approx (\mathbf{B} \otimes \mathbf{A}) \mathbf{G} \mathbf{C}^T$, 其中 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 是三个模型载体, 在不失一般性的基础上假设三者正交, \mathbf{G} 是 TUCKER3 核张量 $\underline{\mathbf{G}}$ 的矩阵形式。核张量的非零元素决定了 $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 列之间的相互关系。相应的模型拟合问题通常由交替最小方差方法求解。TUCKER3 模型可以向量化表示为:

$$\text{vec}(\mathbf{X}) \approx (\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A}) \text{vec}(\mathbf{G})$$

PARAFAC: 当核张量 $\underline{\mathbf{G}}$ 被约束成对角阵的形式 [当 $m \neq l$ 或者 $n \neq l$ 时, $\underline{\mathbf{G}}(l, m, n) = 0$] 可建立 PARAFAC 模型, 有时也被称为规范分解 (CANDECOMP), 或者简称 CP (CANDECOMP-PARAFAC)。PARAFAC 可表示为简洁的矩阵形式 $\mathbf{X} \approx (\mathbf{B} \odot \mathbf{A}) \mathbf{C}^T$, 其中 \odot

为 Khatri-Rao 积。PARAFAC 在某种程度上是最基本的张量模型，因为其与张量秩和低秩分解、低秩拟合间存在直接关系。特别地，利用 Khatri-Rao 的性质，可得

$$\underline{X} \approx (\underline{B} \odot \underline{A}) \underline{C}^T$$

进而

$$\text{vec}(\underline{X}) \approx (\underline{C} \odot \underline{B} \odot \underline{A}) \mathbf{1}$$

其中 $\mathbf{1}$ 是一个全一向量。同样，

$$\underline{X} = \sum_{f=1}^F \underline{a}_f \circ \underline{b}_f \circ \underline{c}_f$$

其中 $\underline{a}_f, \underline{b}_f, \underline{c}_f$ 分别是 $\underline{A}, \underline{B}, \underline{C}$ 的第 f 列。我们考虑一个秩为 F 的 $I \times J \times K$ 维的张量 \underline{X} 。一个 $I \times J \times K$ 维、 F 元、用 $(\underline{A}, \underline{B}, \underline{C})$ 表示的平行因子模型可以被写成 $IJK \times 1$ 阶的向量形式，即 $\underline{x} = (\underline{A} \odot \underline{B} \odot \underline{C}) \mathbf{1}$ ，其中 \underline{A} 为 $(I \times F)$ 维， \underline{B} 为 $(J \times F)$ 维， \underline{C} 为 $(K \times F)$ 维。对于给定的 \underline{X} ($\Leftrightarrow \underline{x}$)，如果 $k_A + k_B + k_C \geq 2F + 2$ ，则 $(\underline{A}, \underline{B}, \underline{C})$ 对于列交换和尺度变换是唯一的，即 $\underline{x} = (\underline{A} \odot \underline{B} \odot \underline{C}) \mathbf{1} \Rightarrow \bar{\underline{A}} = \underline{A} \Pi \underline{A}_a, \bar{\underline{B}} = \underline{B} \Pi \underline{A}_b, \bar{\underline{C}} = \underline{C} \Pi \underline{A}_c$ ，其中 Π 是一个列模糊矩阵， $\underline{A}_a, \underline{A}_b, \underline{A}_c$ 是非奇异对角矩阵并且满足 $\underline{A}_a \underline{A}_b \underline{A}_c = \underline{I}$ 。

当一个待处理的张量 \underline{X} 太大而不能放进内存时，一种合理的方法是尝试将 \underline{X} 压缩成一个小得多的张量，常用的压缩算法是低正交 TUCKER3 分解，已有的 PARAFAC 压缩算法，包括 COMFAC^[23]等。

5.2.2 PARAFAC 压缩感知框架

1. PARAFAC 压缩感知理论

假设将 \underline{x} 压缩成 $\underline{y} = \underline{S} \underline{x}$ ，其中 \underline{S} 为 $d \times IJK$ 维并且 $d \ll IJK$ 。特殊地，我们提出一个压缩矩阵 $\underline{S} = \underline{U}^T \otimes \underline{V}^T \otimes \underline{W}^T$ ，其对应于分别从 I 维、 J 维、 K 维用 $\underline{U}^T, \underline{V}^T, \underline{W}^T$ 乘以 \underline{X} 的各个切片，其中 $\underline{U}, \underline{V}, \underline{W}$ 分别是 $I \times L, J \times M, K \times N$ 维，并且 $L \leq I, M \leq J, N \leq K, LMN \leq IJK$ 。这样的 \underline{S} 很自然地对应于单独压缩的各个维度，当张量是稀疏的时，可以有效地实现相应的乘法运算。由 Kronecker 积的性质，有

$$(\underline{U}^T \otimes \underline{V}^T \otimes \underline{W}^T)(\underline{A} \odot \underline{B} \odot \underline{C}) = ((\underline{U}^T \underline{A}) \odot (\underline{V}^T \underline{B}) \odot (\underline{W}^T \underline{C}))$$

由此可得

$$\underline{y} = ((\underline{U}^T \underline{A}) \odot (\underline{V}^T \underline{B}) \odot (\underline{W}^T \underline{C})) \mathbf{1} = (\tilde{\underline{A}} \odot \tilde{\underline{B}} \odot \tilde{\underline{C}}) \mathbf{1}$$

即压缩后的数据符合一个 $L \times M \times N$ 维、 F 元、由 $(\tilde{\underline{A}}, \tilde{\underline{B}}, \tilde{\underline{C}})$ 规定的 PARAFAC 模型，其中 $\tilde{\underline{A}} = \underline{U}^T \underline{A}, \tilde{\underline{B}} = \underline{V}^T \underline{B}, \tilde{\underline{C}} = \underline{W}^T \underline{C}$ ，压缩图如图 5.2.1 所示。我们有以下结论。

定理 5.2.1^[22]：令 $\underline{x} = (\underline{A} \odot \underline{B} \odot \underline{C}) \mathbf{1} \in \mathbf{R}^{IJK}$ ，其中 \underline{A} 为 $(I \times F)$ 维， \underline{B} 为 $(J \times F)$ 维， \underline{C} 为 $(K \times F)$ 维。将 \underline{A} 压缩至 $\underline{y} = ((\underline{U}^T \underline{A}) \odot (\underline{V}^T \underline{B}) \odot (\underline{W}^T \underline{C})) \mathbf{1} = (\tilde{\underline{A}} \odot \tilde{\underline{B}} \odot \tilde{\underline{C}}) \mathbf{1} \in \mathbf{R}^{LMN}$ ，其中维度

压缩矩阵 $U(I \times L, L \leq I), V(J \times M, M \leq J), W(K \times N, N \leq K)$ 分别从 $\mathbf{R}^L, \mathbf{R}^M, \mathbf{R}^N$ 上关于 Lebesgue 测度的绝对连续分布中随机得到。假设 A, B, C 的列均为稀疏的, 令 n_a, n_b, n_c 分别表示 A, B, C 各列中非零元素数的上界。若 $\min(L, k_A) + \min(M, k_B) + \min(N, k_C) \geq 2F + 2$ 且 $L \geq 2n_a, M \geq 2n_b, N \geq 2n_c$, 则几乎可以肯定, A, B, C 在压缩数据 y 中是可辨识的 (存在列模糊和尺度模糊), 即若 $((U^T A) \odot (V^T B) \odot (W^T C)) \mathbf{1} = y$, 则 $\bar{A} = A \Pi A_a, \bar{B} = B \Pi A_b, \bar{C} = C \Pi A_c$ 以概率 1 成立, 其中 Π 是一个列模糊矩阵, A_a, A_b, A_c 是非奇异对角矩阵并且满足 $A_a A_b A_c = I$ 。

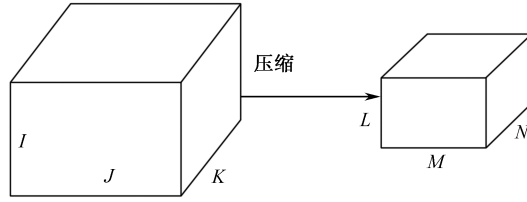


图 5.2.1 PARAFAC 压缩

为了证明该定理, 引入两条引理。

引理 5.2.1^[22]: 设 $\tilde{A} = U^T A$, 其中 A 为 $I \times F$ 维。 $I \times L$ 矩阵 U 从一个在 \mathbf{R}^L 上关于 Lebesgue 测度的绝对连续分布中随机得到 (比如, 有非奇异协方差矩阵的多元高斯函数), 则 $k_{\tilde{A}} = \min(L, k_A)$ 以概率 1 几乎确定成立。

证明: 由 Sylvester 不等式可得 $k_{\tilde{A}}$ 不大于 $\min(L, k_A)$ 。令 $k = \min(L, k_A)$ 。除了一个零测度组合之外所有的 U^T 都可以使 \tilde{A} 的任意 k 列线性独立。任选 \tilde{A} 中的 k 列可以写成 $\tilde{A}_s = U^T A_s$, 其中 A_s 表示 A 中的相应列。设 $\tilde{A}_{s,t}$ 为一 $k \times k$ 、包含 A_s 顶部行的子方阵, 并且 $\tilde{A}_{s,t} = U_t^T A_s$, 其中 U_t^T 包含 U 的顶部 k 行。 $\det(\tilde{A}_{s,t})$ 是 U_t^T 中元素 (实际上是多元多项式) 的解析函数, 一个不是所有取值都为零的解析函数在定义域上的所有值都几乎非零^[24]。为证明对几乎每个 U^T 都有 $\det(\tilde{A}_{s,t}) \neq 0$, 只要找到一个 U^T 使得 $\det(\tilde{A}_{s,t}) \neq 0$ 即可。为达到这个目的, 注意到因为 $k \leq k_A$, A_s 是列满秩的, 并且其值为 k , 因此 A_s 有一个包含 k 个线性独立的行的子集。令 U_t^T 中对应的 k 列组成一个 $k \times k$ 的单位矩阵, 然后将 U_t^T 的剩余元素置零。对于这个特定的 U_t^T 来说, $\det(\tilde{A}_{s,t}) \neq 0$ 。这表明对于除了零测度之外所有的 U^T , \tilde{A} 的任意 k 列都是线性独立的。从 F 列中选出 k 列总共有 $\binom{F}{k}$ 种组合方式, 每种组合都排除了一个测度为零的组合。具有有限个零测度组合的闭集其测度也为零, 因此所有可能的由 \tilde{A} 的任意 k 列构成的子集都几乎确定是线性独立的。

下一个引理是广为人知的压缩感知论文[25]中的理论, 虽然一般并不表述成 k -秩的形式。

引理 5.2.2^[22]: 设 $\tilde{A} = U^T A$, 其中 \tilde{A}, U 都是给定的, A 是待求解的。假设 A 的每一列

至多有 n_a 个非零元素, 并且 $k_{U^T} \geq 2n_a$ (若 $I \times L$ 的矩阵 U 是从关于 \mathbf{R}^L 的 Lebesgue 测度的绝对连续分布中随机得到的, 并且 $\min(I, L) \geq 2n_a$, 则 $k_{U^T} \geq 2n_a$ 以概率 1 成立), 则 A 存在每列至多有 n_a 个非零元素的特解。

我们现在可以证明定理 5.2.1。

证明: 由引理 5.2.1 和压缩张量 $\mathbf{y} = (\tilde{A} \odot \tilde{B} \odot \tilde{C})\mathbf{1}$ 满足 Kruskal 条件可得: $\tilde{A} = U^T A, \tilde{B} = V^T B, \tilde{C} = W^T C$ 是唯一的, 存在列的模交换和尺度变换, 即 $\tilde{A}\Pi A_a, \tilde{B}\Pi B_b, \tilde{C}\Pi C_c$ 是可辨识的, 其中 Π 是一个列模糊矩阵, A_a, B_b, C_c 是非奇异对角矩阵并且满足 $A_a A_b A_c = I$ 。然后由引理 5.2.2 可得 A, B, C 都一定可以分别从 $\tilde{A}\Pi A_a, \tilde{B}\Pi B_b, \tilde{C}\Pi C_c$ 恢复 (存在列的模交换和尺度变换)。

注 5.2.1: 定理 5.2.1 不要求 L, M 或 N 大于等于 F 。若 $L \geq k_A, M \geq k_B, N \geq k_C$, 由定理 5.2.1 可得在相同的 k -秩条件下, A, B, C 都可以分别从压缩数据 \mathbf{y} 中得到 (列的模交换和尺度变换), 就如同我们从未压缩数据 \mathbf{x} 中得到一样。如果忽略了 \mathbf{x} 中潜在的低秩 (多线性、Khatri-Rao) 结构并且试图依照一个有至多 $F n_a n_b n_c$ 个非零元素的稀疏张量来恢复 \mathbf{x} , 则 $LMN \geq 2F n_a n_b n_c$ 的条件是必须要满足的。对于 $n_a = n_b = n_c = n$, $L = M = N$, 并且 $F \leq 2n$, 后者需要 $2Fn^3$ 个采样 (作为对比, 定理 5.2.1 需要 $8n^3$ 个)。

注 5.2.2: 最佳 PARAFAC 拟合是 NP 难的, 但是在实际中交替最小方差 (ALS) 算法可以实现令人满意的准确逼近, 在原空间需要 $O(IJKF)$ 的复杂度, 在压缩空间则需要 $O(LMNF)$ (假设存在一个对总循环次数的限制)。计算一个 Q 个未知数的方程组的最小 ℓ_1 范数解需要最多 $O(Q^{3.5})$ 的计算复杂度。将一个 PARAFAC 模型拟合到压缩数据, 然后对 A, B, C 的每一列求解一个 ℓ_1 最小化的问题总共需要 $O(LMNF + (I^{3.5} + J^{3.5} + K^{3.5})F)$ 的计算量。在非压缩数据域这些并不需要计算, 这对于不能装入内存的大数据来说相当重要。首先利用稀疏性, 然后再在原空间拟合 PARAFAC 模型需要 $O(IJKF + (IJK)^{3.5})$ 的计算量。

下面我们将证明, 如果一个维度没有在 F 元下压缩, 比如 $N \geq F$, 则通过对另外两个维度进行更高程度的压缩 (更小的 L, M), 保证可辨识性是可能的。在下文中, 我们为简便起见, 认为压缩矩阵是满足独立高斯同分布的。

定理 5.2.2^[22]: 令 $\mathbf{x} = (A \odot B \odot C)\mathbf{1} \in \mathbf{R}^{IJK}$, 其中 A 为 $(I \times F)$ 维, B 为 $(J \times F)$ 维, C 为 $(K \times F)$ 维。将 A 压缩至 $\mathbf{y} = ((U^T A) \odot (V^T B) \odot (W^T C))\mathbf{1} = (\tilde{A} \odot \tilde{B} \odot \tilde{C})\mathbf{1} \in \mathbf{R}^{LMN}$, 其中维度压缩矩阵 $U (I \times L, L \leq I), V (J \times M, M \leq J), W (K \times N, N \leq K)$ 的元素是独立标准高斯同分布的。假设 A, B, C 的列是稀疏的, 令 $n_a(n_b, n_c)$ 为 $A(B, C)$ 中每列非零元素数的上界。若 $r_A = r_B = r_C = F$, $L(L-1)M(M-1) \geq 2F(F-1)$, $N \geq F$, 并且 $L \geq 2n_a, M \geq 2n_b, N \geq 2n_c$, 则 A, B, C 几乎肯定可以从压缩数据 \mathbf{y} 中辨识得到 (存在列模糊和尺度模糊)^[28]。

注意到定理 5.2.2 允许三个维度中的两个的压缩低至 \sqrt{F} 阶。证明过程需要以下引理。

引理 5.2.3^[22]: 令 $\tilde{A} = U^T A$, 其中 $A (I \times F)$ 维是确定的高矩阵或者方矩阵 ($I \geq F$) 并且列满秩 ($r_A = F$), 并且 $U (I \times L)$ 维的元素是独立标准高斯同分布的随机变量, 则 \tilde{A}

的分布是关于 \mathbf{R}^{LF} 中的 Lebesgue 测度呈绝对连续的（非奇异多元高斯分布）。

证明：定义 $\mathbf{z} := \text{vec}(\tilde{\mathbf{A}}^T)$ 和 $\mathbf{u} := \text{vec}(\mathbf{U})$ 。则 $\mathbf{z} = \text{vec}(\mathbf{A}^T \mathbf{U} \mathbf{I}) = (\mathbf{I} \otimes \mathbf{A}^T) \text{vec}(\mathbf{U}) = (\mathbf{I} \otimes \mathbf{A}^T) \mathbf{u}$ ，因此 $\mathbf{R}_{\mathbf{z}} := E[\mathbf{z} \mathbf{z}^T] = (\mathbf{I} \otimes \mathbf{A}^T) E[\mathbf{u} \mathbf{u}^T] (\mathbf{I} \otimes \mathbf{A}) = (\mathbf{I} \otimes \mathbf{A}^T) (\mathbf{I} \otimes \mathbf{A}) = \mathbf{I} \otimes (\mathbf{A}^T \mathbf{A})$ ，这其中我们用到了向量化和 Kronecker 积的混合乘积性质。Kronecker 积的秩是秩的乘积，因此 $r_{\mathbf{R}_{\mathbf{z}}} = LF$ 。

现在我们证明定理 5.2.2。

证明：由文献[26]，我们可知，当 $\tilde{\mathbf{A}}, \tilde{\mathbf{B}}$ 是由关于 $\mathbf{R}^{(L+M)F}$ 中的 Lebesgue 测度呈绝对连续的分布中随机得到的， $\tilde{\mathbf{C}}$ 是列满秩的，并且 $L(L-1)M(M-1) \geq 2F(F-1)$ 时，PARAFAC 是几乎确定可辨识的。 $\tilde{\mathbf{C}}$ 的列满秩由引理 5.2.1 可以确定成立。引理 5.2.3 和 \mathbf{U}, \mathbf{V} 的独立性表明 $\tilde{\mathbf{A}}, \tilde{\mathbf{B}}$ 的联合分布是关于 $\mathbf{R}^{(L+M)F}$ 中的 Lebesgue 测度呈绝对连续的。

定理 5.2.1 和定理 5.2.2 可以很容易地推广到四阶或者更高阶的张量（有任意的维数）中。作为示例，应用文献 [27] 中的广义 Kruskal 条件得到。

定理 5.2.3^[22]：令 $\mathbf{x} = (\mathbf{A}_1 \odot \cdots \odot \mathbf{A}_\delta) \mathbf{1} \in \mathbf{R}^{\Pi_{d=1}^\delta L_d}$ ，其中 \mathbf{A}_d 是 $I_d \times F$ 维的。将 \mathbf{x} 压缩至 $\mathbf{y} = (\mathbf{U}_1^T \otimes \cdots \otimes \mathbf{U}_\delta^T) \mathbf{x} = ((\mathbf{U}_1^T \mathbf{A}_1) \odot \cdots \odot (\mathbf{U}_\delta^T \mathbf{A}_\delta)) \mathbf{1} \in \mathbf{R}^{\Pi_{d=1}^\delta L_d}$ ，其中维度压缩矩阵 \mathbf{U}_d ($I_d \times L_d$, $L_d \leq I_d$) 是从关于 $\mathcal{H}^{I_d L_d}$ 中的 Lebesgue 测度绝对连续分布中随机得到的。假设 \mathbf{A}_d 的列是稀疏的，令 n_d 为 \mathbf{A}_d 中每列非零元素数的上界。若 $\sum_{d=1}^\delta \min(L_d, k_{\mathbf{A}_d}) \geq 2F + \delta - 1$ 并且 $\forall d, L_d \geq 2n_d$ ，则原始因子承载 $\{\mathbf{A}_d\}_{d=1}^\delta$ 几乎确定可以从压缩数据 \mathbf{y} 中辨识得到（存在列模糊和尺度模糊）。

2. PARAFAC 压缩感知设计方案 1

以三线模型为例，存在一个三线模型，其承载矩阵为 $\mathbf{A} \in \mathbb{C}^{I \times F}$ 、 $\mathbf{B} \in \mathbb{C}^{J \times F}$ 和 $\mathbf{C} \in \mathbb{C}^{K \times F}$ 。存在三个降维矩阵 $\mathbf{U} \in \mathbb{C}^{I' \times I}$ 、 $\mathbf{V} \in \mathbb{C}^{J' \times J}$ 和 $\mathbf{W} \in \mathbb{C}^{K' \times K}$ ($I' < I, J' < J, K' < K$)，使得

$$\mathbf{A}' = \mathbf{U} \mathbf{A}, \quad \mathbf{B}' = \mathbf{V} \mathbf{B}, \quad \mathbf{C}' = \mathbf{W} \mathbf{C}$$

将 $\mathbf{X}_k = \mathbf{A} \mathbf{D}_k(\mathbf{C}) \mathbf{B}^T, k=1, \dots, K$ 重构矩阵 $\mathbf{X}^{(I \times JK)} = [\mathbf{X}_1 \quad \mathbf{X}_2 \quad \cdots \quad \mathbf{X}_K]$ 。 $\mathbf{X}^{(I \times JK)}$ 为三线模型的矩阵展开形式，对 $\mathbf{X}^{(I \times JK)}$ 进行下矩阵操作：

$$\mathbf{X}^{(I' \times J' K')} = \mathbf{U} \mathbf{X}^{(I \times JK)} (\mathbf{W}^H \otimes \mathbf{V}^H)$$

将 $\mathbf{X}^{(I' \times J' K')}$ 矩阵分割如下： $\mathbf{X}^{(I' \times J' K')} = [\mathbf{X}'_1 \quad \mathbf{X}'_2 \quad \cdots \quad \mathbf{X}'_{K'}]$ ， $\mathbf{X}'_k \in \mathbb{C}^{I' \times J'}$ 可表示为

$$\mathbf{X}'_k = \mathbf{A}' \mathbf{D}_k(\mathbf{C}') \mathbf{B}'^T, k=1, \dots, K'$$

上式为压缩的平行因子模型的切片形式。通过压缩，平行因子维数可大大降低，从而减小平行因子分解复杂度和存储量。对压缩的平行因子模型进行三线分解或平行因子分解，得到

$$\begin{aligned} \hat{\mathbf{A}}' &= \mathbf{A}' \Pi \mathbf{A}_1 + \mathbf{N}_1 = \mathbf{U} \mathbf{A} \Pi \mathbf{A}_1 + \mathbf{N}_1 \\ \hat{\mathbf{B}}' &= \mathbf{B}' \Pi \mathbf{A}_2 + \mathbf{N}_2 = \mathbf{V} \mathbf{B} \Pi \mathbf{A}_2 + \mathbf{N}_2 \\ \hat{\mathbf{C}}' &= \mathbf{C}' \Pi \mathbf{A}_3 + \mathbf{N}_3 = \mathbf{W} \mathbf{C} \Pi \mathbf{A}_3 + \mathbf{N}_3 \end{aligned}$$

其中, Π 为列模糊矩阵; Δ_1 、 Δ_2 和 Δ_3 为尺度模糊矩阵, 而且 $\Delta_1\Delta_2\Delta_3 = \mathbf{I}$, 其中 \mathbf{I} 为单位阵。 \mathbf{N}_1 、 \mathbf{N}_2 和 \mathbf{N}_3 为估计误差。尺度模糊可通过归一化来消除。最后利用稀疏性来恢复信号。该平行因子压缩感知的框架如图 5.2.2 所示。此框架有利于降维矩阵 \mathbf{U} 、 \mathbf{V} 和 \mathbf{W} 的设计, 而且恢复问题可并行处理, 同时该框架很容易推广到四线性模型/多线性模型的压缩感知理论。

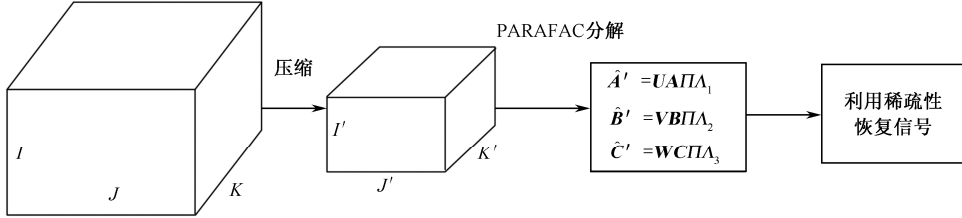


图 5.2.2 平行因子压缩感知的框架 1（三线性模型）

3. PARAFAC 压缩感知设计方案 2

该方案中第一步和第二步与上面方案相同, 将平行因子分解和稀疏性恢复合并到一起。定义 \mathbf{a}_k 为矩阵 \mathbf{A} 的第 k 列, 假设矢量 \mathbf{a}_k ($k=1, \dots, K$) 为稀疏的。在压缩平行因子模型中, 构建以下目标函数:

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \sum_{k=1}^{K'} \frac{1}{2} \left\| \tilde{\mathbf{X}}'_k - \mathbf{A}' \mathbf{D}_k(\mathbf{C}') \mathbf{B}'^T \right\|_F^2 + \sum_{k=1}^K \lambda_k \|\mathbf{a}_k\|_0 \\ \text{st. } \mathbf{A}' = \mathbf{U}\mathbf{A} \\ \mathbf{B}' = \mathbf{V}\mathbf{B} \\ \mathbf{C}' = \mathbf{W}\mathbf{C} \end{aligned}$$

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \sum_{k=1}^{K'} \frac{1}{2} \left\| \tilde{\mathbf{X}}'_k - \mathbf{A}' \mathbf{D}_k(\mathbf{C}') \mathbf{B}'^T \right\|_F^2 + \sum_{k=1}^K \lambda_k \|\mathbf{a}_k\|_1 \\ \text{st. } \mathbf{A}' = \mathbf{U}\mathbf{A} \\ \mathbf{B}' = \mathbf{V}\mathbf{B} \\ \mathbf{C}' = \mathbf{W}\mathbf{C} \end{aligned}$$

其中, $\|\cdot\|_0$ 、 $\|\cdot\|_1$ 和 $\|\cdot\|_F$ 为 l_0 范数、 l_1 范数和 F 范数。 $\tilde{\mathbf{X}}'_k$ 为含噪信号。我们也可探索其他一些目标函数。通过上式目标函数优化恢复信号。该平行因子压缩感知的框架如图 5.2.3 所示。上式可推广到四线性分解、多线性分解或其他承载矩阵稀疏性约束的情况。

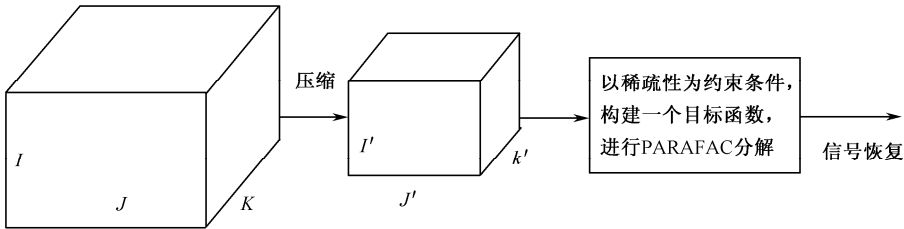


图 5.2.3 平行因子压缩感知的框架 2（三线性模型）

5.2.3 平行因子模型填充

平行因子模型出现数据缺省情况下，平行因子理论会失效，不能进行信号检测和参数估计。这就需要通过一些方法或算法将平行因子模型中这些空缺的元素准确地填充，这就对应平行因子模型填充问题。

1. 平行因子模型填充方案 1

该方案将平行因子模型分解成多个切片形式，利用矩阵秩稀疏性和矩阵填充理论来填充每个切片，进而填充整个平行因子模型；最后利用平行因子分解实现信号检测和参数估计。此方法简单易行。假设 $\underline{\mathbf{X}}$ 为缺省的平行因子模型， $\mathbf{X}_k \in \mathbb{C}^{I \times J}$ 为平行因子模型 $\underline{\mathbf{X}}$ 第 k 切片，填充后为 $\mathbf{Y}_k \in \mathbb{C}^{I \times J}$ 。于是可以通过以下优化问题来实现矩阵填充：

$$\min_{\mathbf{Y}_k} \frac{1}{2} \|P_{\Omega}(\mathbf{Y}_k - \mathbf{X}_k)\|_{\text{F}}^2 + \lambda \text{rank}(\mathbf{Y}_k), \quad k=1, \dots, K$$

其中， Ω 是已知元素下标的集合。 $P_{\Omega}()$ 表示一种投影算子，它将矩阵在 Ω 以外的元素置 0， Ω 内部元素保持不变。 $\text{rank}(\cdot)$ 为矩阵的秩，这是个 NP 难题。用矩阵的奇异值的和，即核范数，来近似地替代矩阵的秩^[72]：

$$\min_{\mathbf{Y}_k} \frac{1}{2} \|P_{\Omega}(\mathbf{Y}_k - \mathbf{X}_k)\|_{\text{F}}^2 + \lambda \|\mathbf{Y}_k\|_*, \quad k=1, \dots, K$$

其中， $\|\mathbf{Y}_k\|_* = \sum_{i=1}^n \sigma_i(\mathbf{Y}_k)$ ， $\sigma_i(\mathbf{Y}_k)$ 表示矩阵 $\mathbf{Y}_k \in \mathbb{C}^{I \times J}$ 的第 i 个的奇异值， $n = \max(I, J)$ 。核范数对应于矩阵奇异值组成的向量的 l_1 范数，转化为凸优化问题。在此方面，本项目组做了一定的基础工作，可推广到平行因子模型填充问题研究中。

2. 平行因子模型填充方案 2

该方案将接收信号构建为三线性/多线性模型，其中部分元素是缺省的；利用平行因子秩稀疏性，进行平行因子模型填充；最后利用平行因子分解实现参数估计或信号检测。

将信号稀疏性定义中矩阵的秩稀疏性，扩展到平行因子模型的秩稀疏；利用平行因子模型低秩性，进行平行因子模型填充。以三线性模型为例，接收缺省平行因子模型信号 $\underline{\mathbf{X}}$ ， $x_{i,j,k}$ 为其中 (i,j,k) 的元素。 $\underline{\mathbf{Y}}$ 为填充后的平行因子模型。于是可以通过如下优化问题来实现平行因子模型填充：

$$\min_{\underline{\mathbf{Y}}} \frac{1}{2} \|P_{\Omega}(\underline{\mathbf{Y}} - \underline{\mathbf{X}})\|_{\text{F}}^2 + \|\underline{\mathbf{Y}}\|_*$$

其中， $\|\underline{\mathbf{Y}}\|_* = \sum_{k=1}^K \lambda_k \|\mathbf{Y}_k\|_*$ ， \mathbf{Y}_k 是 $\underline{\mathbf{Y}}$ 第 k 个切片。此问题还可表示为：

$$\min_{\underline{\mathbf{Y}}} \frac{1}{2} \|P_{\Omega}(\underline{\mathbf{Y}} - \underline{\mathbf{X}})\|_{\text{F}}^2 + \sum_{k=1}^K \lambda_k \|\mathbf{Y}_k\|_*$$

平行因子模型填充后再进行阵列信号检测和参数估计。这里研究缺省元素可辨识性分析。

3. 平行因子模型填充方案 3

该方案将接收信号构建成三线性/多线性模型，其中部分元素是缺省的；将平行因子填充和平行因子分解相结合。以平行因子承载矩阵的秩稀疏性为约束，构建一个目标函数，进行平行因子模型填充的同时实现平行因子分解；最后实现参数估计或信号检测。

以三线性模型为例子，三线性模型可表示为 $\underline{\mathbf{X}} = \sum_{f=1}^F \mathbf{a}_f \circ \mathbf{b}_f \circ \mathbf{c}_f$ ，其中， \circ 为外积， $\mathbf{a}_f, \mathbf{b}_f$ 和 \mathbf{c}_f 为承载矩阵 \mathbf{A}, \mathbf{B} 和 \mathbf{C} 中 f 列。通过优化以下问题来实现平行因子填充和平行因子分解的联合处理：

$$\begin{aligned} \min_{\underline{\mathbf{Y}}, \mathbf{A}, \mathbf{B}, \mathbf{C}} \quad & \left\| \underline{\mathbf{Y}} - \sum_{f=1}^F \mathbf{a}_f \circ \mathbf{b}_f \circ \mathbf{c}_f \right\|_F^2 + \mu \left\| P_{\Omega}(\underline{\mathbf{Y}} - \underline{\mathbf{X}}) \right\|_F^2 \\ \text{st.} \quad & \left\| \mathbf{A} \right\|_* \leq F \\ & \left\| \mathbf{B} \right\|_* \leq F \\ & \left\| \mathbf{C} \right\|_* \leq F \end{aligned}$$

5.3 本章小结

本章研究了压缩感知基本原理，并给出了平行因子压缩感知框架。将传统基于矩阵的压缩感知理论扩展到了高阶张量压缩感知理论。本章我们参考了很多文献，如文献[15,20,22]，在此对相关作者表示感谢。对于平行因子压缩感知理论应用将在第 9 章详细说明。

参考文献

- [1] Donoho D L. Compressed sensing [J]. IEEE Transactions on Information Theory, 2006, 52 (4) : 1289-1306.
- [2] Candès E J. Compressive sampling [C]. Proceedings of the International Congress of Mathematicians, Madrid, Spain, 2006: 1433-1452.
- [3] Candès E J, Tao T. Near-optimal signal recovery from random projections: Universal encoding strategies? [J]. IEEE Transactions on Information Theory, 2006, 52 (12) : 5406-5425.
- [4] Candès E J, Romberg J K, Tao T. Stable signal recovery from incomplete and inaccurate measurements [J]. Communications on Pure and Applied Mathematics, 2006, 59 (8) : 1207-1223.
- [5] Babadi B, Kalouptsidis N, Tarokh V. Asymptotic achievability of the Cramér–Rao bound

- for noisy compressive sampling [J]. IEEE Transactions on Signal Processing, 2009, 57 (3) : 1233-1236.
- [6] Sarvotham S, Baron D, Baraniuk R G. Measurements vs. bits: Compressed sensing meets information theory [C]. Proceedings of 44th Allerton Conference on Communication, Control, and Computing, 2006: 123-130.
- [7] Goyal V K, Fletcher A K, Rangan S. Compressive sampling and lossy compression [J]. IEEE Signal Processing Magazine, 2008, 25 (2) : 48-56.
- [8] aubock G, Hlawatsch F. A compressed sensing technique for OFDM channel estimation in mobile environments: Exploiting channel sparsity for reducing pilots[C]. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Las Vegas, NV, 2008: 2885-2888.
- [9] Wang Z, Arce G R, Paredes J L, et al. Compressed detection for ultra-wideband impulse radio[C]. IEEE 8th Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Helsinki, 2007: 1-5.
- [10] Haiping Y, Shaohua W, Qinyu Z, et al. A Compressed Sensing Approach for IR-UWB Communication[C]. International Conference on Multimedia and Signal Processing (CMSP), Guilin, China, 2011, 2: 3-7.
- [11] Liu T C K, Xiaodai D, Wu-Sheng L. Compressed Sensing Maximum Likelihood Channel Estimation for Ultra-Wideband Impulse Radio[C]. IEEE International Conference on Communications, Dresden, 2009: 1-5.
- [12] Paredes J L, Arce G R, Wang Z. Ultra-Wideband Compressed Sensing: Channel Estimation[J]. IEEE Journal of Selected Topics in Signal Processing, 2007, 1(3): 383-395.
- [13] Le T N, Jaewoon K, Yoan S. An improved ToA estimation in compressed sensing-based UWB systems[C]. IEEE International Conference on Communication Systems (ICCS), Singapore, 2010: 249-253.
- [14] Shaohua W, Qinyu Z, Haiping Y, et al. High-resolution TOA estimation for IR-UWB ranging based on low-rate compressed sampling[C]. 6th International ICST Conference on Communications and Networking in China (CHINACOM), Harbin, 2011: 478-483.
- [15] 林波. 基于压缩感知的辐射源 DOA 估计. 国防科学技术大学硕士学位论文, 2011.
- [16] 李树涛, 魏丹. 压缩传感综述. 自动化学报, 2009, 35 (11) :1369-1377.
- [17] E. Candès, J. Romberg, T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information[C]. IEEE Trans. Information Theory. 2006, 52 (2) : 489-509.
- [18] Donoho D L, Elad M, Temlyakov V N. Stable recovery of sparse overcomplete representations in the presence of noise[C]. IEEE Transactions on Information Theory, 2006, 52 (1) :6-18.

- [19] Recht B, Fazel M, Parrilo P A. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 2010, 52 (3) : 471–501.
- [20] 彭义刚, 索津莉, 戴琼海, 等. 从压缩传感到低秩矩阵恢复: 理论与应用. *自动化学报*, 2012, 38 (12) : 1-11.
- [21] Candès E J, Tao T. The power of convex relaxation: near-optimal matrix completion. *IEEE Transactions on Information Theory*, 2009, 56 (5) : 2053–2080.
- [22] Sidiropoulos, N. D. Kyrillidis, A.: 'Multi-Way Compressed Sensing for Sparse Low-Rank Tensors', *IEEE Signal Processing Letters*, 2012, 19, (11) , pp.757-760.
- [23] R. Bro, N. Sidiropoulos, and G. Giannakis, "A fast least squares algorithm for separating trilinear mixtures," in *Proc. ICA99 Int. Work-shop on Independent Component Analysis and Blind Signal Separation*, 1999, pp. 289–294.
- [24] T. Jiang, N. Sidiropoulos, and J. ten Berge, "Almost sure identifiability of multidimensional harmonic retrieval," *IEEE Trans. Signal Process.* , 2001, 49 (9) :1849–1859.
- [25] D. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via minimization," *Proc. Nat. Acad. Sci.*, 2003, 100 (5) : 2197–2202.
- [26] Stegeman, J. ten Berge, and L. De Lathauwer, "Sufficient conditions for uniqueness in CANDECOMP/PARAFAC and INDSCAL with random component matrices," *Psychometrika*, 2006, 71 (2) : 219–229.
- [27] T. Jiang and N. Sidiropoulos, "Kruskal's permutation lemma and the identification of CANDECOMP/ PARAFAC and bilinear models with constant modulus constraints," *IEEE Trans. Signal Process.* , 2004, 52 (9) : 2625–2636.

第6章

三线性分解在通信和信号处理中的应用

本章就三线性分解理论在通信和信号处理中的应用给出几个例子。将自适应 PARAFAC 应用于 MIMO 雷达系统中，实现单基地 MIMO 雷达角度跟踪；同时提出一种非圆（NC）-PARAFAC 算法，该算法可以用于任意声矢量阵列下非圆信号源的 2D-DOA 估计。

6.1 多天线 OFDM 系中一种基于三线性分解盲载波频偏估计算法

本节研究多天线 OFDM 系统的载波频偏（Carrier Frequency Offset, CFO）估计问题，并提出一种基于三线性分解的 CFO 估计算法^[1]。与借助旋转不变性进行信号参数估计（Estimating Signal Parameters via Rotational Invariance Techniques, ESPRIT）算法和循环统计量（Cyclo-Stationarity, CS）算法相比，该算法提高了 CFO 的估计性能。更重要的是，该算法在没有虚载波的情况下依然能够正常工作。仿真结果显示了该算法的性能。

6.1.1 数据模型

考虑一个有 I 根接收天线的单输入多输出（Single Input Multiple Output, SIMO）OFDM 系统上行，假设每根接收天线的 CFO 相同。在这个 OFDM 系统的 N 个子载波中， P 个子载波用来进行数据传输， $N - P$ 个子载波是虚载波。循环前缀（Cyclic Prefix, CP）采用 L 个采样间隔，其中 L 大于最大传输延迟。加入 CP 后，输出信号通过多径衰落信道传输。

定义 $H_i(n) = \sum_{l=0}^{L_m-1} h_i(l) e^{-j2\pi nl/N}$ 作为第 i 根天线对应的第 n 个子载波的信道频率响应，其中 $\{h_i(l)\}_{l=0}^{L_m-1}$ 是离散信道冲击响应。第 i 根接收天线的频域信道向量是 $\mathbf{h}_i = [H_i(1), H_i(2), \dots, H_i(P)]^T$ 。因此，多天线接收机的频域信道矩阵是

$$\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_I]^T \in \mathbb{C}^{I \times P} \quad (6.1.1)$$

移除 CP 后，第 i 根接收天线的输出信号可以定义为

$$\mathbf{x}_i(k) = \mathbf{E} \mathbf{F}_p \text{diag}(\mathbf{h}_i) \mathbf{s}(k) e^{j2\pi \Delta f (k-1)(N+L)} \quad (6.1.2)$$

其中 $\mathbf{s}(k) = [s_1(k), s_2(k), \dots, s_P(k)]^T$ 代表第 k 块， Δf 是 CFO， $\mathbf{E} = \text{diag}\{1, e^{j2\pi \Delta f}, \dots, e^{j(N-1)2\pi \Delta f}\} \in \mathbb{C}^{N \times N}$ 代表 CFO 矩阵， $\mathbf{F}_p \in \mathbb{C}^{N \times P}$ 代表逆离散傅立叶变换矩阵的前 P 列。假设 K 块的信道参数是常量，信源矩阵可以被定义为 $\mathbf{S} = [\mathbf{s}(1), \mathbf{s}(2), \dots, \mathbf{s}(K)]^T \in \mathbb{C}^{K \times P}$ 。定义 $\mathbf{X}_i = [\mathbf{x}_i(1), \mathbf{x}_i(2), \dots, \mathbf{x}_i(K)]$ ，则可以得到

$$\mathbf{X}_i = \mathbf{A} \text{diag}(\mathbf{h}_i) \mathbf{B}^T = \mathbf{A} \mathbf{D}_i(\mathbf{H}) \mathbf{B}^T, \quad i = 1, \dots, I \quad (6.1.3)$$

其中 $\mathbf{B} = \text{diag}\{1, e^{j2\pi \Delta f (N+L)}, \dots, e^{j2\pi \Delta f (K-1)(N+L)}\} \mathbf{S} \in \mathbb{C}^{K \times P}$ ， $\mathbf{D}_i(\cdot)$ 是抽取矩阵的第 i 行并将其构造为一个对角矩阵。Vandermonde 矩阵 \mathbf{A} 是

$$\mathbf{A} = \mathbf{E} \mathbf{F}_p = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{j2\pi(\frac{0}{N} + \Delta f)} & e^{j2\pi(\frac{1}{N} + \Delta f)} & \dots & e^{j2\pi(\frac{P-1}{N} + \Delta f)} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j2\pi(N-1)(\frac{0}{N} + \Delta f)} & e^{j2\pi(N-1)(\frac{1}{N} + \Delta f)} & \dots & e^{j2\pi(N-1)(\frac{P-1}{N} + \Delta f)} \end{bmatrix} \in \mathbb{C}^{N \times P} \quad (6.1.4)$$

由于含噪，接收信号模型变为 $\tilde{\mathbf{X}}_i = \mathbf{A} \mathbf{D}_i(\mathbf{H}) \mathbf{B}^T + \mathbf{W}_i$ ， $i = 1, \dots, I$ ，其中 \mathbf{W}_i 是第 i 根天线的接收噪声。公式 (6.1.3) 中的信号可以定义为三线性模型^[2]

$$x_{n,k,i} = \sum_{p=1}^P a_{n,p} b_{k,p} h_{i,p}, \quad n=1, \dots, N, k=1, \dots, K, i=1, \dots, I \quad (6.1.5)$$

其中 $h_{i,p}$ 代表矩阵 \mathbf{H} 的第 (i, p) 个元素，其余同理。公式 (6.1.3) 可以看成三线性模型沿着天线方向的切片。同时，公式 (6.1.5) 中三线性模型的对称性允许其他形式的矩阵重构，因此得到 $\mathbf{Y}_n = \mathbf{B} \mathbf{D}_n(\mathbf{A}) \mathbf{H}^T$ ， $n=1, \dots, N$ ，以及 $\mathbf{Z}_k = \mathbf{H} \mathbf{D}_k(\mathbf{B}) \mathbf{A}^T$ ， $k=1, \dots, K$ 。下面的章节将提出如何利用三线性分解进行 CFO 估计，并推导出基于三线性分解进行 CFO 估计的算法。

6.1.2 算法原理

本节利用三线性分解来估计矩阵 \mathbf{A} ，然后用最小二乘法 (Least Square, LS) 获得 CFO 估计。

1. 三线性分解

三线性分解最小二乘法 (Trilinear Alternating Least Square, TALS) 是基于三线性模型的一种数据检测方法，TALS 的原理是在有噪情况下拟合三线性模型。TALS 算法的基本流

程是：

- ① 在获得矩阵前一次估计值的基础上每次用 LS 更新一个矩阵；
- ② 对其他矩阵进行更新；
- ③ 重复上述过程直到 LS 代价函数收敛。

TALS 算法的细节描述如下。

公式 (6.1.3) 的信号也可以改写为下式：

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_I \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{H}) \\ \mathbf{A}\mathbf{D}_2(\mathbf{H}) \\ \vdots \\ \mathbf{A}\mathbf{D}_I(\mathbf{H}) \end{bmatrix} \mathbf{B}^T = [\mathbf{H} \odot \mathbf{A}] \mathbf{B}^T \quad (6.1.6)$$

其中 $\mathbf{H} \odot \mathbf{A}$ 代表 Khatri-Rao 积。LS 拟合为 $\min_{\mathbf{A}, \mathbf{H}, \mathbf{B}} \|\tilde{\mathbf{X}} - [\mathbf{H} \odot \mathbf{A}] \mathbf{B}^T\|_F$ ，其中 $\tilde{\mathbf{X}}$ 是含噪信号。 \mathbf{B} 矩阵的 LS 更新是

$$\hat{\mathbf{B}}^T = [\hat{\mathbf{H}} \odot \hat{\mathbf{A}}]^\dagger \tilde{\mathbf{X}} \quad (6.1.7)$$

其中 $\hat{\mathbf{A}}$ 和 $\hat{\mathbf{H}}$ 分别是前一次 \mathbf{A} 和 \mathbf{H} 的估计。

相似地，按照第 2 种方法切片得到 $\mathbf{Y}_n = \mathbf{B}\mathbf{D}_n(\mathbf{A})\mathbf{H}^T$ ， $n = 1, 2, \dots, N$ ，由此得到

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_N \end{bmatrix} = \begin{bmatrix} \mathbf{B}\mathbf{D}_1(\mathbf{A}) \\ \mathbf{B}\mathbf{D}_2(\mathbf{A}) \\ \vdots \\ \mathbf{B}\mathbf{D}_N(\mathbf{A}) \end{bmatrix} \mathbf{H}^T = [\mathbf{A} \odot \mathbf{B}] \mathbf{H}^T \quad (6.1.8)$$

根据公式 (6.1.8)，LS 拟合是 $\min_{\mathbf{A}, \mathbf{H}, \mathbf{B}} \|\tilde{\mathbf{Y}} - [\mathbf{A} \odot \mathbf{B}] \mathbf{H}^T\|_F$ ，其中 $\tilde{\mathbf{Y}}$ 是含噪信号。那么矩阵 \mathbf{H} 的 LS 更新是

$$\hat{\mathbf{H}}^T = [\hat{\mathbf{A}} \odot \hat{\mathbf{B}}]^\dagger \tilde{\mathbf{Y}} \quad (6.1.9)$$

其中 $\hat{\mathbf{A}}$ 和 $\hat{\mathbf{B}}$ 分别为前一次 \mathbf{A} 和 \mathbf{B} 的估计。

最后，按照第 3 种切片方法得到 $\mathbf{Z}_k = \mathbf{H}\mathbf{D}_k(\mathbf{B})\mathbf{A}^T$ ， $k = 1, 2, \dots, K$ ，矩阵 \mathbf{A} 的 LS 更新是

$$\hat{\mathbf{A}}^T = [\hat{\mathbf{B}} \odot \hat{\mathbf{H}}]^\dagger \tilde{\mathbf{Z}} \quad (6.1.10)$$

其中 $\hat{\mathbf{H}}$ 和 $\hat{\mathbf{B}}$ 分别是前一次 \mathbf{H} 和 \mathbf{B} 的估计，且 $\tilde{\mathbf{Z}}$ 为含噪信号。

根据式 (6.1.7)、式 (6.1.9) 和式 (6.1.10) 可知，矩阵 \mathbf{B} 、 \mathbf{H} 和 \mathbf{A} 分别通过 LS 算法更新。对于零均值高斯白噪声，TALS 满足最大似然 (Maximal Likelihood, ML) 估计并使得算法性能达到全局最小变量的要求^[3]。TALS 算法的主要缺点是收敛速度慢^[4]，尽管 TALS 算法很容易实现而且保证能够收敛，该算法通常用随机初始化或者用特征值分解方法初始化来加快算法的收敛。本章利用复平行因子分析 (Complex Parallel Factor Analysis, COMFAC) 算法^[5]进行三线性分解。COMFAC 算法是一种快速的 TALS 算法，该算法可以加快 LS 拟合。作为一种改进，COMFAC 算法将三维数据压缩为一个规模更小的三维数据。在这个压缩空间中对模型进行拟合后，再回到原有空间，这只需少量 TALS 迭代即可收敛。

2. 可辨识性

定理 6.1.1 ^[6]: $Y_n = BD_n(A)H^T$, $n = 1, \dots, N$, 其中 $A \in \mathbb{C}^{N \times P}$, $H \in \mathbb{C}^{I \times P}$, $B \in \mathbb{C}^{K \times P}$ 。考虑到这些矩阵是满秩的, 且满 k -秩, 并且矩阵 A 具有 Vandermonde 特性, 如果

$$\min(P, K) + \min(N + \min(I, P), 2P) \geq 2P + 2 \quad (6.1.11)$$

那么 A 、 H 和 B 是可辨识的, 也就是说, 对于其他构造 Y_n , $n = 1, \dots, N$, 的矩阵 \hat{A} 、 \hat{H} 、 \hat{B} , 可以通过 $\hat{A} = A\Pi\Delta_1 + N_1$ 、 $\hat{H} = H\Pi\Delta_2 + N_2$ 、 $\hat{B} = B\Pi\Delta_3 + N_3$ 与 A 、 H 和 B 联系起来, 其中 Π 是排列矩阵, N_1 、 N_2 、 N_3 为误差, 而 Δ_1 、 Δ_2 、 Δ_3 是对角矩阵且满足 $\Delta_1\Delta_2\Delta_3 = I_P$ 。

一般情况下有 $P > I$, 那么公式 (6.1.11) 变成 $\min(P, K) + \min(N + I, 2P) \geq 2P + 2$ 。当 $N + I > 2P$ 且 $K < P$ 时, 约束条件改写为 $2 \leq K < P$ 。因此, 三线性分解算法能够支持小样本。当 $K > P$ 且 $P \geq 2$ 时, 约束条件变成 $N + I > P + 2$ 。

对于接收到的含噪信号, 利用三线性分解来获得估计矩阵 $\hat{A} = A\Pi\Delta_1 + N$, 其中 N 是噪声矩阵。由三线性分解的性质可知, 估计矩阵存在列模糊和尺度模糊的情况。然而值得注意的是, 尺度模糊可以很容易解决, 且列模糊并不影响 CFO 的估计性能。

3. CFO 估计

如同在 6.1.1 节讨论的那样, 矩阵 A 的第 P 列是 $a_p(\Delta f) = [1, e^{j2\pi(\frac{p-1}{N} + \Delta f)}, \dots, e^{j2\pi(N-1)(\frac{p-1}{N} + \Delta f)}]^T$, 定义

$$g = \text{imag}(\ln(a_p(\Delta f))) \quad (6.1.12)$$

其中 $\ln(\cdot)$ 是自然对数, $\text{imag}(\cdot)$ 是取复数的虚部。

$$g = [0, 2\pi(\frac{p-1}{N} + \Delta f), \dots, 2\pi(N-1)(\frac{p-1}{N} + \Delta f)]^T = (\frac{p-1}{N} + \Delta f)q$$

其中 $q = [0, 2\pi, \dots, 2\pi(N-1)]^T$, 那么 LS 准则就可以用来估计 CFO Δf 。用 \hat{a}_l (矩阵 \hat{A} 的第 l 列) 中每一个元素除以 $\hat{a}_{l,1}$ (向量 \hat{a}_l 的第 1 个元素) 的方法可以解决尺度模糊的问题, 并且根据公式 (6.1.12), \hat{g} 可以通过这个过程获得。现在采用 LS 准则来估计 CFO。

定义 LS 拟合为 $Qc = \hat{g}$, 其中 $Q = [1_{N \times 1}, q]$, $c = [c_0, f_p]^T$ 且 f_p 是 $\frac{p-1}{N} + \Delta f$, $p \in \{1, 2, \dots, P\}$ 的估计值。最后可以得到 c 的 LS 解

$$\begin{bmatrix} \hat{c}_0 \\ \hat{f}_p \end{bmatrix} = Q^+ \hat{g} \quad (6.1.13)$$

类似地, 我们可以估计 \hat{f}_p , $p = 1, 2, \dots, P$ 。因此, CFO 可以通过下式估计出来:

$$\Delta \hat{f} = \frac{1}{P} \left(\sum_{p=1}^P \hat{f}_p - \sum_{p=1}^P \frac{p-1}{N} \right) \quad (6.1.14)$$

值得注意的是, 由于在 CFO 估计中使用了求和操作 $\sum_{p=1}^P \hat{f}_p$, CFO 估计性能不会被列模糊影响。

对比 ESPRIT 算法, 三线性分解算法的复杂度更高。ESPRIT 算法的复杂度为 $O(KIN^2 + N^3 + 2(N-1)P^2 + 3P^3)$ 。在三线性分解算法中, 每次 TALS 迭代的复杂度是 $O(3P^3 + 3PNIK + P^2(IN + IK + NK + I + N + K))$, 并且由于使用 COMFAC 算法, 大量的迭代被用于三线性分解中。

值得注意的是, 基于三线性分解的算法不能用于 SISO-OFDM 系统。当接收机只包含 1 根天线或者 $I=1$ 时, 信号变为 $\mathbf{X}=\mathbf{A}\mathbf{D}_1(\mathbf{H})\mathbf{B}^T=\mathbf{A}\mathbf{S}$, 其中 $\mathbf{S}=\mathbf{D}_1(\mathbf{H})\mathbf{B}^T$ 。考虑 $\mathbf{X}=\mathbf{A}\mathbf{S}$ 作为线性模型, 其中矩阵 \mathbf{A} 具有恒模特性。众所周知, 线性分解缺乏唯一解, 然而, 当具备恒模特性时, 线性分解是唯一的^[7,8]。

6.1.3 仿真结果

在仿真中, 考虑子载波数 $N=32$ 且 CP 的采样间隔 $L=8$ 的 SIMO-OFDM 系统。CFO $\Delta\omega=2\pi\Delta f$ 固定在 0.3ω , 其中 $\omega=2\pi/N$ 代表子载波间隔。现在得到有噪情况下的接收信号 $\tilde{\mathbf{X}}_i=\mathbf{A}\mathbf{D}_i(\mathbf{H})\mathbf{B}^T+\mathbf{W}_i$ ($i=1,\dots,I$), 其中 \mathbf{W}_i 是加性高斯白噪声矩阵。定义 SNR:

$$\text{SNR}=10\log_{10}\frac{\sum_{i=1}^I\|\mathbf{A}\mathbf{D}_i(\mathbf{H})\mathbf{B}^T\|_{\text{F}}^2}{\sum_{i=1}^I\|\mathbf{W}_i\|_{\text{F}}^2}\text{dB} \quad (6.1.15)$$

为了定量评估 CFO 估计性能, 使用 1000 次 Monte Carlo 仿真来估计 MSE。定义 MSE 为 $\text{MSE}=\frac{1}{1000}\sum_{m=1}^{1000}(\frac{\hat{\Delta f}^m-\Delta f}{1/N})^2$, 其中 $\hat{\Delta f}^m$ 是第 m 次 Monte Carlo 仿真得到的 CFO 估计值, 并且 Δf 是 CFO 的精确值。

发射机和第 i 根接收天线间的信道用 $h_i(t)=\sum_{l=0}^{L_m-1}\rho_{il}\delta(t-\tau_{il})$ 进行建模, 其中 L_m 是多径数, ρ_{il} 和 τ_{il} 分别是复增益和第 l 条路径的延迟。信道由 4 条独立的瑞利衰落信道构成, 多径路径功率服从指数分布。每次 Monte Carlo 仿真中, 所有的信道参数都随机生成。

首先研究仿真中三线性分解算法的收敛性。三线性拟合的残差平方和 (Sum of Squared Residuals, SSR) 可以定义为 $\text{SSR}=\sum_{n=1}^N\sum_{k=1}^K\sum_{i=1}^I[\tilde{x}_{n,k,i}-\sum_{p=1}^P\hat{a}_{n,p}\hat{b}_{k,p}\hat{h}_{i,p}]^2$, 其中 $\tilde{x}_{n,k,i}$ 是含噪数据。

定义 $\text{DSSR}=\text{SSR}_i-\text{SSR}_0$, 其中 SSR_i 是第 i 次迭代的 SSR, SSR_0 是 SSR 的收敛条件。图 6.1.1 展示了 $N=32$ 、 $P=20$ 、 $I=4$ 、 $K=100$ 和 $\text{SNR}=16\text{dB}$ 条件下 COMFAC 算法和 TALS 算法的收敛性能, 其中 N 、 P 、 I 和 K 分别是总载波数、传输数据的子载波数、天线数和信号块数。由图 6.1.1 可知, COMFAC 的收敛性能比 TALS 好很多。

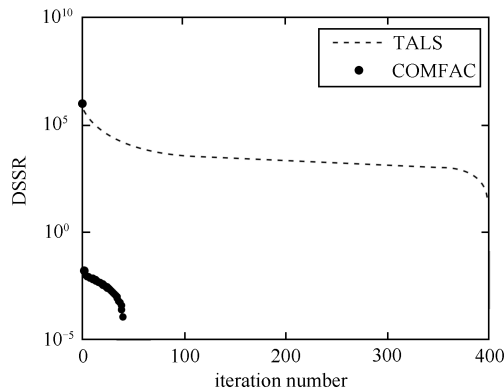


图 6.1.1 算法收敛性能

将三线性分解算法与 ESPRIT 算法和 CS 算法比较, 这两种算法在 SIMO-OFDM 系统

中都能正常工作。图 6.1.2 展示了在 $N=32$ 、 $P=20$ 、 $I=4$ 和 $K=50$ 或 $K=200$ 情况下的 CFO 估计性能比较。由图 6.1.2 可以看到，相比于 ESPRIT 算法和 CS 算法，三线性分解算法性能更加优越。并且可以很清楚地看到 CS 算法的性能比其他算法都差，所以接下来就不再对此算法进行仿真了。总而言之，三线性分解算法的 CFO 估计性能在这三种算法中是最好的。

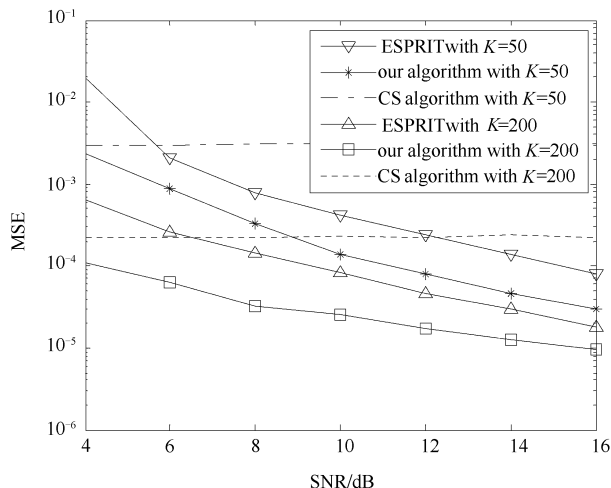


图 6.1.2 $N=32$, $P=20$, $I=4$ 算法估计性能比较

图 6.1.3 展示了 $N=32$ 、 $P=20$ 、 $I=4$ 且 $K=100$ 时的误码率 (Bit-Error-Rate, BER) 性能以及 CFO 估计算法补偿 CFO 后的 BER 性能。为了验证算法在 CFO 估计时的性能，假设信道状况是已知的。采用二进制相移键控 (Binary Phase Shift Keying, BPSK) 调制信号，且进行 5000 次 Monte Carlo 仿真并收集 10 个数据块来描绘 BER。从图 6.1.3 中可以看出，三线性分解算法的 BER 性能比 ESPRIT 算法好。

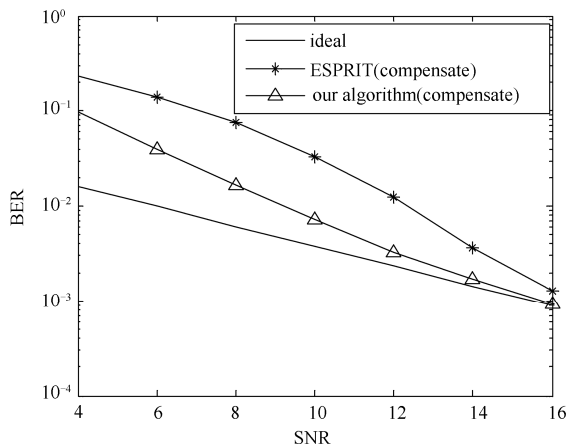


图 6.1.3 BER 性能比较

图 6.1.4 展示了 $N = 32$ 、 $I = 4$ 、 $P = 20$ 且 K 取不同值时所提算法的 CFO 估计性能。通过图 6.1.4 可以看到，CFO 估计性能随着 K 的增加而提高了。这是因为当 K 增加时，算法对于噪声的抑制能力提高了。

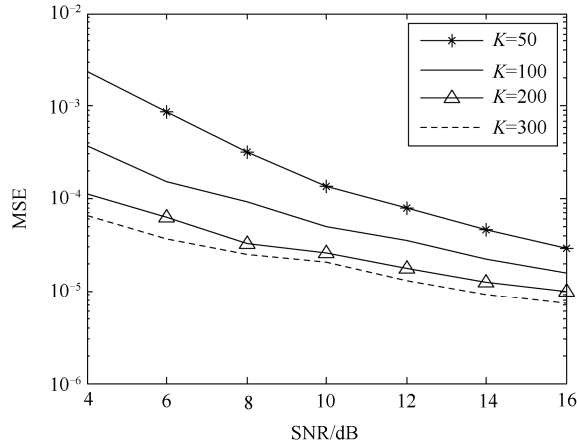


图 6.1.4 K 取不同的值时 CFO 估计性能

图 6.1.5 展示了 $N = 32$ 、 $K = 100$ 、 $P = 20$ 且 I 取不同值时所提算法的 CFO 估计性能。通过图 6.1.5 可以看到，由于接收分集增益，多接收天线提高了 CFO 估计性能。因此，随着天线数的增加，算法的 CFO 估计性能也逐渐提高。

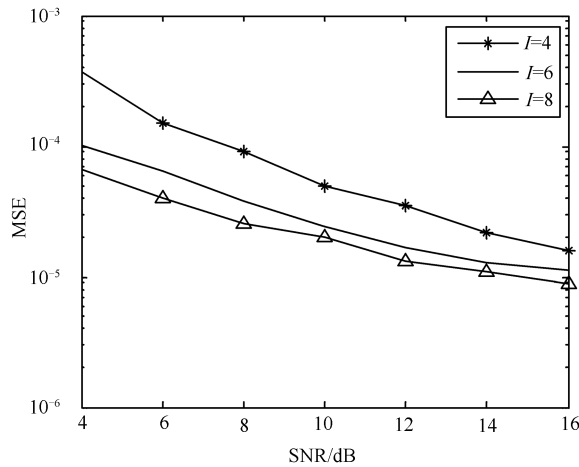


图 6.1.5 I 取不同的值时 CFO 估计性能

图 6.1.6 展示了 $N=32$ 、 $I=4$ 、 $K=100$ 且 P 取不同值时的 CFO 估计性能。通过图 6.1.6 可以看到，随着 P 的增加 CFO 估计性能下降了。因为当 P 增加时，矩阵 $\mathbf{A} \in \mathbb{C}^{N \times P}$ 的秩增加了，这使得矩阵 \mathbf{A} 的估计性能下降，所以 CFO 估计性能也下降了。特别地，当 $P = 32$ 时，

OFDM 系统中不存在虚载波, 此时一些依赖于虚载波的盲 CFO 估计算法 (例如 MUSIC 算法和 ESPRIT 算法) 就不能工作了。因此, 三线性分解算法有一个很明显的优势, 即可以在没有虚载波的情况下正常工作。

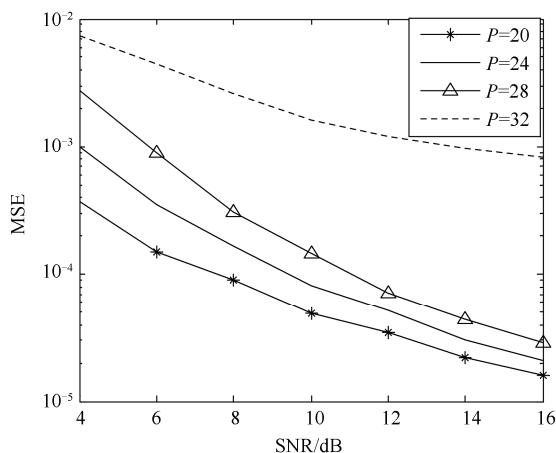


图 6.1.6 P 取不同的值时 CFO 估计性能

图 6.1.7 展示了 $K=100$ 、 $I=4$ 且 N 取不同值时的 CFO 估计性能比较。通过图 6.1.7 可以看到, 随着 N 的增加, CFO 估计性能也下降了。这时因为当 N 成倍增加时 (取 16、32 和 64), $P=5N/8$ 也相应扩大, 则矩阵 $\mathbf{A} \in \mathbb{C}^{N \times P}$ 的秩也变大了。因此, CFO 估计性能随着矩阵 \mathbf{A} 的估计性能的下降也逐渐降低。然而, 即使 N 取得再大, 只要满足约束条件, 三线性分解算法依然能够正常工作。例如, 当 $N=1024$ 、 $P=640$ 且 $I=4$ 时, 约束条件 $K \geq 254$ 。

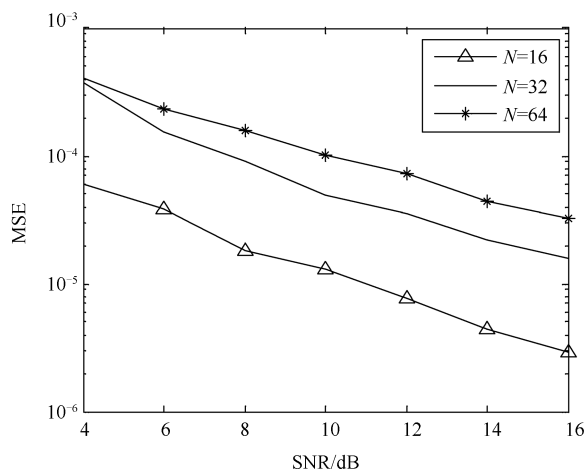


图 6.1.7 N 取不同的值时 CFO 估计性能

最后,图 6.1.8 展示了不同的 CFO 取值时估计性能的比较。仿真参数为 $N=32$ 、 $P=20$ 、 $I=4$ 、 $K=100$ 且 CFO 的范围是 $-0.5\omega \sim 0.5\omega$ 。通过图 6.1.8 可以看到,对于不同的 CFO 取值,通过三线性分解算法进行 CFO 估计的性能非常接近。

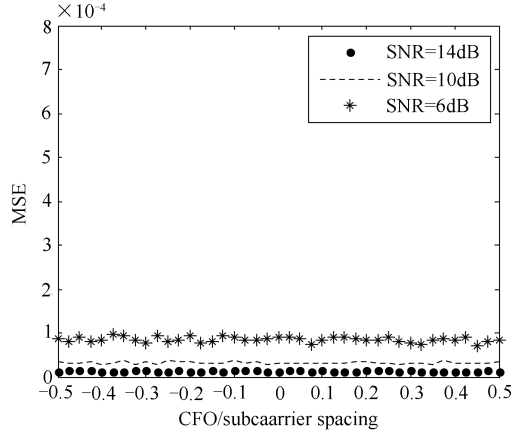


图 6.1.8 CFO 取不同的值时 CFO 估计性能

总而言之,三线性分解算法的优点可以归纳如下:作为一种盲算法,它的性能比 ESPRIT 算法和 CS 算法都好,它不需要信道状态信息,也不需要其他的信源统计特性,更重要的是,三线性分解算法可以在没有虚载波的情况下正常工作。而三线性分解算法的缺点可以归纳为两点:一是复杂度比 ESPRIT 算法高;二是当 $I=1$ (接收机只有 1 根天线) 时,接收信号只能被看成线性模型,这种情况下三线性分解算法不能正常工作。

本节把多天线 OFDM 系统的估计问题和三线性模型联系起来,并推导了一种基于三线性分解的盲 CFO 估计算法,该算法既不需要导频也不需要训练序列。仿真结果显示,三线性分解算法的性能比 ESPRIT 算法和 CS 算法好很多。更重要的是,三线性分解算法不需要恒模特性和其他的信源统计特性,甚至能够在没有虚载波的情况下正常工作。

6.2 基于三线性分解的任意矢量传感器阵的二维波达方向估计

三线性分解可以视为 ESPRIT 算法和联合对角化思想的一种推广,由于其所具有的唯一性及其特点,已被应用到信号处理的很多领域中,具有重要的现实意义。下面将三线性分解方法延用于任意阵型的声矢量传感器阵中,用其对阵列的到达角进行估计^[9]。同时,该算法提供了不知位置的不规则或随机间隔阵列的盲波达方向 (Direction Of Arrival, DOA) 估计。

6.2.1 数据模型

以窄带信号为例, 设有 K 个信号入射到声矢量传感器阵上, 该声矢量传感器阵是由 M 个任意分布在三维空间中的阵元组成的, 如图 6.2.1 所示, 阵列可以是一个不规则的几何阵列。定义第 k 个阵元的位置为 $\mathbf{r}_k = (x_k, y_k, z_k)$ 。考虑入射信号位于远源场, 即信源有足够远使得到达波在本质上超过了阵元长度。假设噪声是独立于信号的加性高斯白噪声, 第 k 个信号的来波方向为 $\boldsymbol{\theta}_k = [\phi_k, \varphi_k]^T$, 其中, φ_k 和 ϕ_k 分别为入射信号的仰角和方位角, 且 $0 \leq \phi_k < 2\pi$, $0 \leq \varphi_k < \pi$ 。则第 k 个入射信号在第 m 个阵元上 t 时刻的输出可表示为

$$\begin{bmatrix} \mathbf{x}_p(t) \\ \mathbf{x}_v(t) \end{bmatrix} = \sum_{k=1}^K \begin{bmatrix} 1 \\ \mathbf{u}_k \end{bmatrix} e^{j2\pi\tau_k} b_k(t) \quad (6.2.1)$$

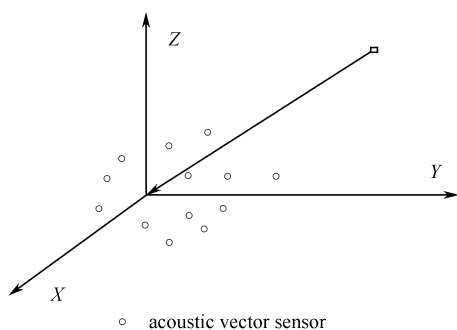


图 6.2.1 声矢量传感器阵的几何形式

其中, $\mathbf{x}_p(t)$ 和 $\mathbf{x}_v(t)$ 分别表示测量信号的压强部分和速率部分, $b_k(t)$ 为第 k 个信源的传输信号, τ_k 为第 k 个源信号投射到第 m 个阵元时, 相对于选定参考点的时延差, \mathbf{u}_k 为传感器中指向信源的单位矢量, 可表示为

$$\mathbf{u}_k = \begin{bmatrix} \cos \phi_k \cos \varphi_k \\ \sin \phi_k \cos \varphi_k \\ \sin \varphi_k \end{bmatrix} \quad (6.2.2)$$

上述阵列的矢量传感器位于 $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_M$ 位置上的输出可表示为

$$\mathbf{x}(t) = \boldsymbol{\Psi}(\boldsymbol{\theta}) \mathbf{b}(t) \quad (6.2.3)$$

其中, $\mathbf{b}(t)$ 包含 K 个信源, $\boldsymbol{\Psi}(\boldsymbol{\theta}) = [\boldsymbol{\psi}(\boldsymbol{\theta}_1) \ \boldsymbol{\psi}(\boldsymbol{\theta}_2) \ \dots \ \boldsymbol{\psi}(\boldsymbol{\theta}_K)]$ 。定义 $\boldsymbol{\psi}(\boldsymbol{\theta}_k) = \mathbf{a}(\boldsymbol{\theta}_k) \otimes \mathbf{h}(\boldsymbol{\theta}_k)$, \otimes 代表 Kronecker 积, $\mathbf{a}(\boldsymbol{\theta}_k)$ 是一个 $M \times 1$ 的声矢量传感器阵的方向矢量, 对于第 k 个信源具有相同的阵列结构。 $\mathbf{h}(\boldsymbol{\theta}_k) = [1 \ \mathbf{u}_k^T]^T$ 表示第 k 个信源的方位矢量。

假设信道的状态信息在 L 个符号长度内是不变的, 则可定义 $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(L)]$, 于是有

$$\mathbf{X} = [\mathbf{A} \odot \mathbf{H}] \mathbf{B}^T \quad (6.2.4)$$

其中, 矩阵 $\mathbf{A} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \dots, \mathbf{a}(\theta_K)] \in \mathbb{C}^{M \times K}$ 为方向矩阵, 矩阵 $\mathbf{H} = [\mathbf{h}(\theta_1), \mathbf{h}(\theta_2), \dots, \mathbf{h}(\theta_K)] \in \mathbb{C}^{4 \times K}$ 为方位矩阵, 矩阵 $\mathbf{B} = [\mathbf{b}(1), \mathbf{b}(2), \dots, \mathbf{b}(L)]^T$ 为信源矩阵, $\mathbf{A} \odot \mathbf{H}$ 表示矩阵 \mathbf{A} 和矩阵 \mathbf{H} 的 Khatri–Rao 积。

式 (6.2.4) 具体可表示为

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_{..1} \\ \mathbf{X}_{..2} \\ \vdots \\ \mathbf{X}_{..M} \end{bmatrix} = \begin{bmatrix} \mathbf{H}\mathbf{D}_1(\mathbf{A}) \\ \mathbf{H}\mathbf{D}_2(\mathbf{A}) \\ \vdots \\ \mathbf{H}\mathbf{D}_M(\mathbf{A}) \end{bmatrix} \mathbf{B}^T \quad (6.2.5)$$

其中, $\mathbf{D}_m(\mathbf{A})$ 表示由矩阵 \mathbf{A} 的第 m 行元素构成的对角矩阵, 即 $\mathbf{D}_m(\mathbf{A}) = \text{diag}([a_{m,1}, a_{m,2}, \dots, a_{m,K}])$, 这里, $a_{m,k}$ 表示矩阵 \mathbf{A} 的第 (m, k) 元素, 则式 (6.2.5) 中的 $\mathbf{X}_{..m}$ 可表示为

$$\mathbf{X}_{..m} = \mathbf{H}\mathbf{D}_m(\mathbf{A})\mathbf{B}^T, \quad m=1, 2, \dots, M \quad (6.2.6)$$

加上噪声, 接收信号模型可表示为 $\tilde{\mathbf{X}}_{..m} = \mathbf{H}\mathbf{D}_m(\mathbf{A})\mathbf{B}^T + \mathbf{V}_{..m}$, $m=1, 2, \dots, M$, 其中, $\mathbf{V}_{..m}$ 是与第 m 个切片相对应的接收噪声, 是一个 $4 \times L$ 维的高斯白噪声矩阵。于是, 可以把式 (6.2.6) 中的信号表示为如下的三线性模型^[2]:

$$x_{m,l,p} = \sum_{k=1}^K a_{m,k} b_{l,k} h_{p,k}, \quad m=1, \dots, M, \quad l=1, \dots, L, \quad p=1, 2, 3, 4 \quad (6.2.7)$$

其中, $b_{l,k}$ 表示矩阵 \mathbf{B} 的第 (l, k) 个元素; 同样地, $a_{m,k}$ 和 $h_{p,k}$ 分别表示矩阵 \mathbf{A} 和矩阵 \mathbf{H} 的第 (m, k) 和第 (p, k) 个元素。可以用三线性模型表示三种不同种类分集, 如图 (6.2.2) 所示。 $\mathbf{X}_{..m} = \mathbf{H}\mathbf{D}_m(\mathbf{A})\mathbf{B}^T$, $m=1, \dots, M$, 可以解释成沿三个不同维度去“切”三维阵列 \mathbf{X} 。同样地, 根据式 (6.2.8) 可以得到以下等式:

$$\mathbf{Y}_{..p} = \mathbf{B}\mathbf{D}_p(\mathbf{H})\mathbf{A}^T, \quad p=1, 2, 3, 4 \quad (6.2.8)$$

$$\mathbf{Z}_{..l} = \mathbf{A}\mathbf{D}_l(\mathbf{B})\mathbf{H}^T, \quad l=1, \dots, L \quad (6.2.9)$$

$\mathbf{X}_{..m}$, $\mathbf{Y}_{..p}$ 和 $\mathbf{Z}_{..l}$ 分别为从不同方向“切”三线性模型时所得到的结果, 如图 6.2.2 所示。

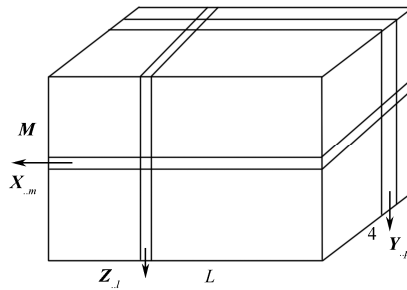


图 6.2.2 三线性模型

6.2.2 三线性分解

本节的方法是首先用三线性分解得到矩阵 \mathbf{H} , 然后估计声矢量传感器阵的二维 DOA。首先, TALS 算法是三线性模型常见的数据检测方法, 其基本原理是: 每一步更新一个矩阵,

对余下的矩阵，依据前一次估计的结果利用最小二乘来更新；继续对其他矩阵进行更新；重复以上步骤直到算法收敛。将式 (6.2.5) 的 LS (最小二乘) 拟合为

$$\min_{A, H, B} \left\| \begin{bmatrix} \tilde{X}_{..1} \\ \tilde{X}_{..2} \\ \vdots \\ \tilde{X}_{..M} \end{bmatrix} - \begin{bmatrix} HD_1(A) \\ HD_2(A) \\ \vdots \\ HD_M(A) \end{bmatrix} B^T \right\|_F \quad (6.2.10)$$

其中， $\tilde{X}_{..m}$ ， $m = 1, \dots, M$ 为含噪信号的切片，对 B 的最小二乘更新为

$$\hat{B}^T = \begin{bmatrix} \hat{H}D_1(\hat{A}) \\ \hat{H}D_2(\hat{A}) \\ \vdots \\ \hat{H}D_M(\hat{A}) \end{bmatrix}^+ \begin{bmatrix} \tilde{X}_{..1} \\ \tilde{X}_{..2} \\ \vdots \\ \tilde{X}_{..M} \end{bmatrix} \quad (6.2.11)$$

其中， \hat{A} 和 \hat{H} 表示上次估计的 A 和 H 的估计。

同样地，从第二个方向的切片： $Y_{..p} = BD_p(H)A^T$ ， $p = 1, 2, 3, 4$ 。于是可以重新写为

$$\begin{bmatrix} Y_{..1} \\ Y_{..2} \\ Y_{..3} \\ Y_{..4} \end{bmatrix} = \begin{bmatrix} BD_1(H) \\ BD_2(H) \\ BD_3(H) \\ BD_4(H) \end{bmatrix} A^T \quad (6.2.12)$$

最小二乘表示为

$$\min_{A, H, B} \left\| \begin{bmatrix} \tilde{Y}_{..1} \\ \tilde{Y}_{..2} \\ \tilde{Y}_{..3} \\ \tilde{Y}_{..4} \end{bmatrix} - \begin{bmatrix} BD_1(H) \\ BD_2(H) \\ BD_3(H) \\ BD_4(H) \end{bmatrix} A^T \right\|_F \quad (6.2.13)$$

其中， $\tilde{Y}_{..p}$ ， $p = 1, \dots, 4$ 为含噪信号的切片。对 A 的最小二乘更新为

$$\hat{A}^T = \begin{bmatrix} \hat{B}D_1(\hat{H}) \\ \hat{B}D_2(\hat{H}) \\ \hat{B}D_3(\hat{H}) \\ \hat{B}D_4(\hat{H}) \end{bmatrix}^+ \begin{bmatrix} \tilde{Y}_{..1} \\ \tilde{Y}_{..2} \\ \tilde{Y}_{..3} \\ \tilde{Y}_{..4} \end{bmatrix} \quad (6.2.14)$$

其中， \hat{B} 和 \hat{H} 表示上次估计的 B 和 H 的估计。

最后，从第三个方向的切片： $Z_{..l} = AD_l(B)H^T$ ， $l = 1, \dots, L$ 。同样地，对 H 的最小二乘更新为

$$\hat{H}^T = \begin{bmatrix} \hat{A}D_1(\hat{B}) \\ \hat{A}D_2(\hat{B}) \\ \vdots \\ \hat{A}D_L(\hat{B}) \end{bmatrix}^+ \begin{bmatrix} \tilde{Z}_{..1} \\ \tilde{Z}_{..2} \\ \vdots \\ \tilde{Z}_{..L} \end{bmatrix} \quad (6.2.15)$$

其中， $\tilde{Z}_{..l}$ ， $l = 1, \dots, L$ 为含噪信号切片。 \hat{B} 和 \hat{A} 表示上次对 B 和 A 的估计。

三线性中的残差平方和可定义为

$$\text{SSR} = \sum_{m=1}^M \sum_{l=1}^L \sum_{p=1}^4 \left| \tilde{x}_{m,l,p} - \sum_{k=1}^K \hat{a}_{m,k} \hat{b}_{l,k} \hat{h}_{p,k} \right|^2 \quad (6.2.16)$$

式中, $\tilde{x}_{m,l,p}$ 是含噪数据, $\hat{a}_{m,k}$ 、 $\hat{b}_{l,k}$ 和 $\hat{h}_{p,k}$ 分别表示 $a_{m,k}$ 、 $b_{l,k}$ 和 $h_{p,k}$ 的估计。

由上述可知, 矩阵 \mathbf{B} 、 \mathbf{A} 和 \mathbf{H} 依据最小二乘分别更新为式 (6.2.11)、式 (6.2.14) 和式 (6.2.15)。一旦 $\text{SSR} \leq 10^{-9}$, 矩阵更新结束。TALS 很容易实施, 且其可随机初始化或利用 EVD 的方法加快收敛。本节将 COMFAC 用于三线性分解, COMFAC 算法本质上是 TALS 的一种快速实现算法, 可以加速最小二乘。

本节中研究的基于三线性分解的声矢量传感器阵的波达角度估计算法的基本思路是: 首先利用接收信号的三线性模型求出估计的矩阵 \mathbf{H} , 然后得到相应的声矢量传感器阵的二维角度估计。

6.2.3 可辨识性和唯一性

三线性模型的本质特征就是其唯一性和可辨识性。即在没有模糊的情况下, \mathbf{A} 、 \mathbf{B} 和 \mathbf{H} 是可辨识的。

定理 6.2.1: 给定 $\mathbf{X}_i = \mathbf{B}\mathbf{D}_i(\mathbf{A})\mathbf{C}^T$, $i=1, \dots, I$, $\mathbf{A} \in \mathbb{C}^{I \times F}$, $\mathbf{B} \in \mathbb{C}^{J \times F}$, $\mathbf{C} \in \mathbb{C}^{K \times F}$, 如果

$$k_A + k_B + k_C \geq 2F + 2 \quad (6.2.17)$$

则 \mathbf{A} 、 \mathbf{B} 和 \mathbf{H} 对于列交换和 (复数) 尺度变换是唯一的。于是, 根据三线性分解可得到以下的分解矩阵:

$$\bar{\mathbf{A}} = \mathbf{A}\mathbf{\Pi}\mathbf{A}_1, \quad \bar{\mathbf{B}} = \mathbf{B}\mathbf{\Pi}\mathbf{A}_2, \quad \bar{\mathbf{H}} = \mathbf{H}\mathbf{\Pi}\mathbf{A}_3 \quad (6.2.18)$$

其中, $\mathbf{\Pi}$ 是列交换矩阵; $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$ 是尺度模糊矩阵, 满足 $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3 = \mathbf{I}_K$ 。

考虑到信号分离的问题, 列交换和尺度模糊是存在的。而且也不难发现, 列交换模糊可以通过先知的或内嵌的信息解决, 而尺度模糊也可以经由增益控制、归一化或内嵌信息消除。

总的来说, 这三个矩阵都满足一个条件, 即: 从绝对连续分布中取出的相对独立的列组成的矩阵具有满 k -秩。所以根据式 (6.2.17) 有

$$\min(4, K) + \min(L, K) + \min(M, K) \geq 2K + 2 \quad (6.2.19)$$

当 $K \geq 4$, $L \geq K$, $M \leq K$ 时, 可辨识的条件变为 $M \geq K - 2$; 当 $2 \leq K \leq 4$, $L \geq K$, $M \leq K$ 时, 可辨识的条件为 $M \geq 2$ 。而且, 当信源数大于声矢量传感器阵的阵元数时 ($K \geq M$), 算法仍具备有效性。

当 $2 \leq K \leq 4$, $L \leq K$, $M \geq K$, 可辨识的充分条件又变为 $L \geq 2$; 当 $K \geq 4$, $L \leq K$, $M \geq K$, 可辨识的充分条件则变为 $L \geq K - 2$ 。而且, 在快拍数很小的情况下, 该算法仍具有较好的估计性能。

6.2.4 算法原理

本节的方法是首先用三线性分解得到矩阵 \mathbf{H} , 然后估计声矢量传感器阵的二维到达角。根据接收的含噪信号, 通过三线性分解可求得 $\hat{\mathbf{H}} = \mathbf{H}\mathbf{\Pi}\mathbf{A}_3 + \mathbf{N}$, $\hat{\mathbf{H}}$ 表示上次得到 \mathbf{H} 的估

计, \mathbf{A}_3 是对角列模糊矩阵。 \mathbf{N} 为估计的误差矩阵。将矩阵 $\hat{\mathbf{H}}$ 归一化得到矩阵 $\hat{\mathbf{H}}$, 矩阵 $\hat{\mathbf{H}}$ 的第 i 列为 $\hat{\mathbf{h}}_i$ 。在没有噪声的情况下, $\hat{\mathbf{h}}_i$ 可表示为

$$\hat{\mathbf{h}}_i = \begin{bmatrix} 1 \\ \cos \phi_i \cos \varphi_i \\ \sin \phi_i \cos \varphi_i \\ \sin \varphi_i \end{bmatrix} \quad (6.2.20)$$

于是可得到二维 DOA 估计:

$$\hat{\phi}_i = \sin^{-1}(\hat{\mathbf{h}}_i(4)) \quad (6.2.21)$$

$$\hat{\phi}_i = \sin^{-1}(\hat{\mathbf{h}}_i(3) / \cos(\hat{\phi}_k)) \quad (6.2.22)$$

式 (6.2.21) 中, $\hat{\phi}_i$ 是 ϕ_i 的估计值; 式 (6.2.22) 中, $\hat{\phi}_i$ 是 ϕ_i 的估计值。

本节分析了基于三线性分解的声矢量传感器阵的二维 DOA 估计方法, 其算法流程如图 6.2.3 所示, 具体步骤如下。

- ① 随机初始化矩阵 \mathbf{A} 、 \mathbf{B} 和 \mathbf{H} ;
- ② 根据式 (6.2.11) 对信源矩阵 \mathbf{B} 进行最小二乘更新;
- ③ 根据式 (6.2.14) 对矩阵 \mathbf{A} 进行最小二乘更新;
- ④ 根据式 (6.2.15) 对矩阵 \mathbf{H} 进行最小二乘更新;
- ⑤ 重复 2~4 步, 直到收敛;
- ⑥ 最终根据估计的矩阵 \mathbf{H} 估计出 DOA。

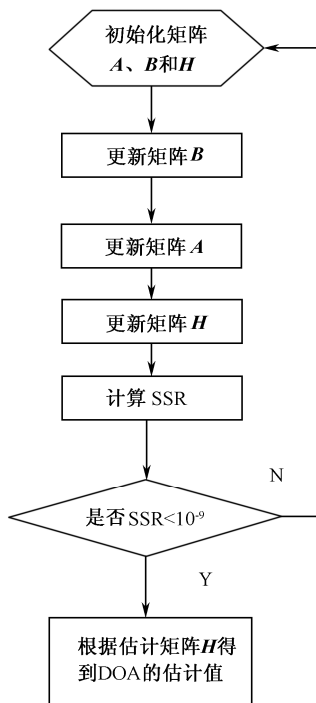


图 6.2.3 算法流程框图

6.2.5 仿真结果

为了能准确地估计出该算法的性能, 本小节将进行 Monte Carlo 仿真, 设其仿真次数为 1000。定义均方根误差(Root Mean Squared Error, RMSE)为 $RMSE = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{1000} \sum_{n=1}^{1000} (\hat{\phi}_k^n - \phi_k)^2}$, 其中, $\hat{\phi}_k^n$ 是第 n 个 Monte Carl 仿真实验的第 k 个信源 DOA ϕ_k 估计值, ϕ_k 是 DOA 的真实值, K 是信源数。

令 $\tilde{\mathbf{X}}_{..m} = \mathbf{H}\mathbf{D}_m(\mathbf{A})\mathbf{B}^T + \mathbf{V}_{..m}$ ($m = 1, 2, \dots, M$) 为接收信号, 其中 $\mathbf{V}_{..m}$ 表示加性白噪声矩阵 (AWGN), 定义信噪比 (SNR) 为

$$SNR = 10 \log_{10} \frac{\sum_{m=1}^M \|\mathbf{H}\mathbf{D}_m(\mathbf{A})\mathbf{B}^T\|_F^2}{\sum_{m=1}^M \|\mathbf{V}_{..m}\|_F^2} \text{ dB} \quad (6.2.23)$$

PARAFAC 算法的角度估计如图 6.2.4~图 6.2.6 所示。

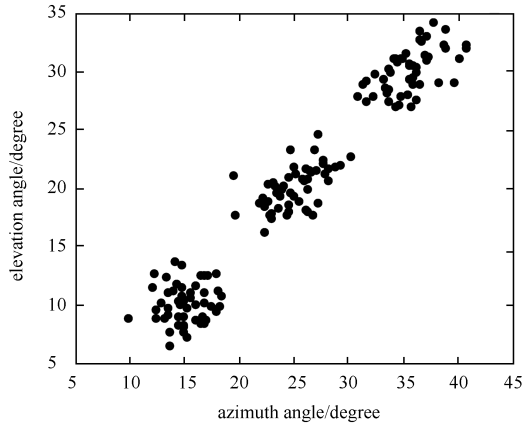


图 6.2.4 PARAFAC 算法的角度估计 (SNR = 0dB)

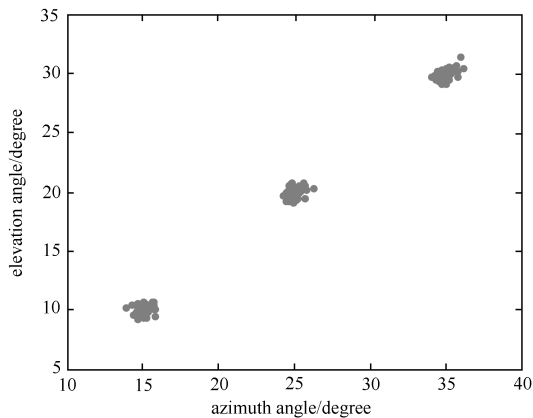
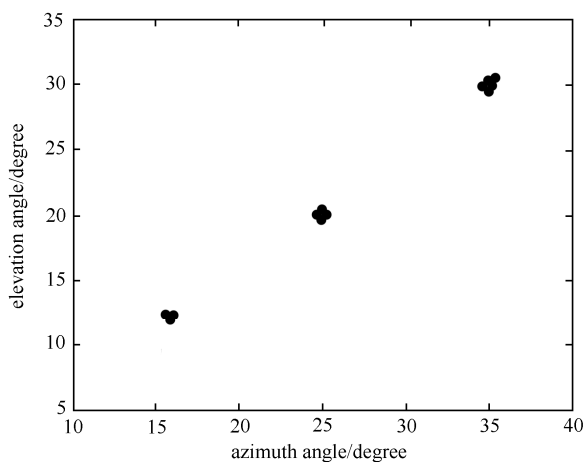
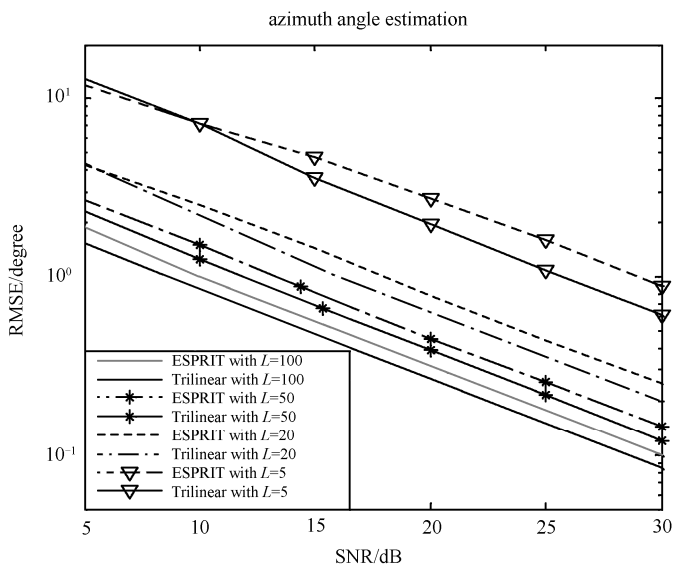


图 6.2.5 PARAFAC 算法的角度估计 (SNR = 15dB)

图 6.2.6 PARAFAC 算法的角度估计 ($\text{SNR} = 30\text{dB}$)

在下面的仿真(图 6.2.7~图 6.2.12)中,假设声矢量传感器阵由 M 个阵元构成,阵元间距为半波长;快拍数为 L ;入射信号的角度分别为 $(15^\circ, 10^\circ)$, $(25^\circ, 20^\circ)$ 和 $(35^\circ, 30^\circ)$;除了图 6.2.7 和图 6.2.8 外,其他仿真图中的非相干信源数均为 $K = 3$ 。

图 6.2.7 不同快拍数 (L) 时方位角的估计性能比较

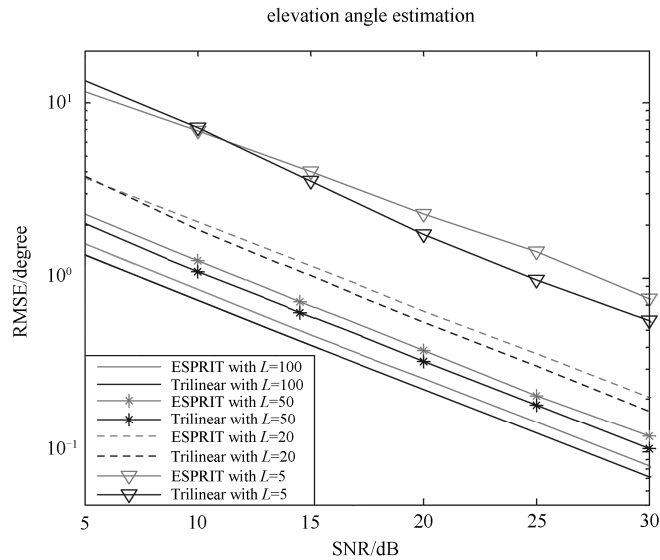


图 6.2.8 不同快拍数 (L) 时仰角的估计性能比较

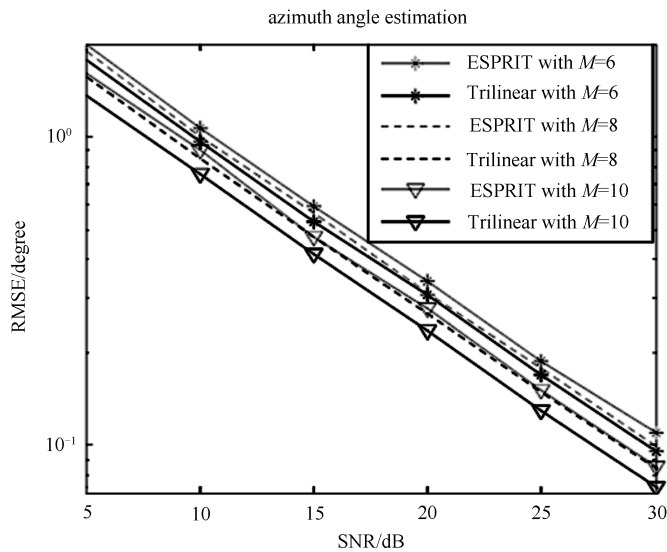
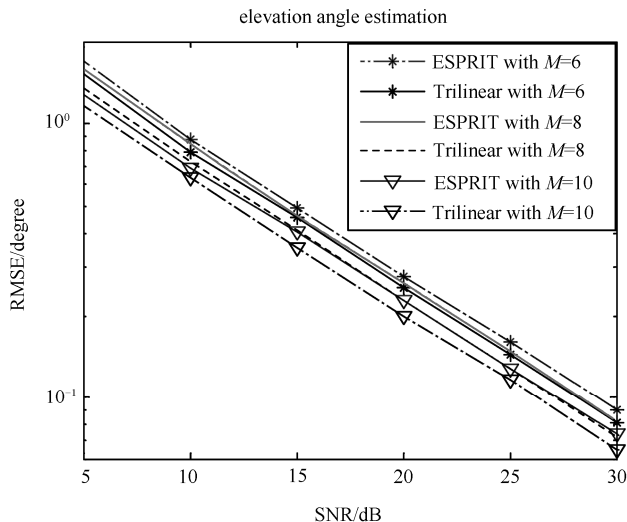
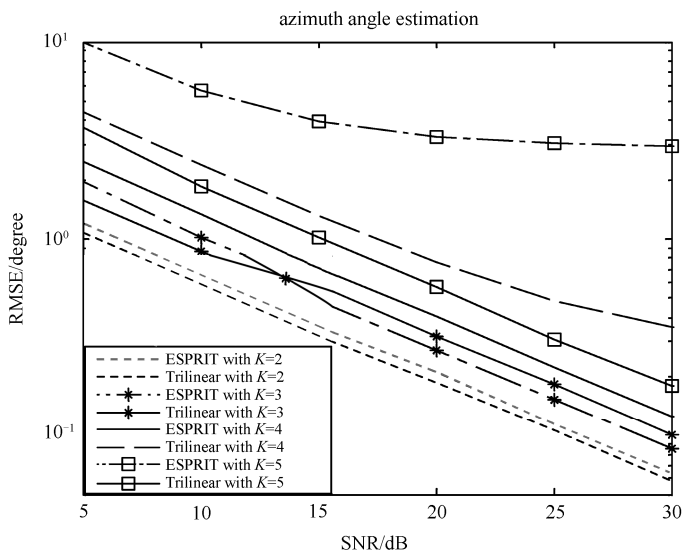


图 6.2.9 不同阵元数 (M) 时方位角的估计性能比较 ($L = 100$)

图 6.2.4 是信源数为 3 时该算法的角度估计 ($M = 8$ 、 $L = 200$ 和 $\text{SNR} = 0\text{dB}$)，接着在其他参数条件不变的情况下，改变信噪比的大小再次对角度进行估计。图 6.2.5 和图 6.2.6 显示了信噪比分别为 $\text{SNR}=15\text{dB}$ 和 $\text{SNR}=30\text{dB}$ 时该算法的角度估计结果。从仿真结果可以看出，该算法可以很准确地估计出到达角。

图 6.2.10 不同阵元数 (M) 时仰角的估计性能比较 ($L = 100$)图 6.2.11 不同信源数 (K) 时方位角的估计性能比较 ($M = 8, L = 100$)

将该算法和 ESPRIT 算法做性能比较, 图 6.2.7 和图 6.2.8 分别显示了快拍数 L 不同时, 这两种算法仰角和方位角的估计性能比较 ($M = 8, K = 3$)。从图 6.2.7 和图 6.2.8 中可以看出, PARAFAC 算法和 ESPRIT 算法的估计性能都随着快拍数 L 的增加而越来越好, 而且通过比较可知, 相同快拍数时 PARAFAC 算法的性能较 ESPRIT 算法好很多。尤其在 $L = 5$ 且高信噪比的情况下, PARAFAC 算法的估计性能要比 ESPRIT 算法高出 2dB。

图 6.2.9 和图 6.2.10 改变了阵元数 M 值, 再次分别比较了该算法和 ESPRIT 算法的仰角和方位角的估计性能 ($L = 100, K = 3$)。从图 6.2.9 和图 6.2.10 中可以很明显地看出, 随

随着阵元数的增加 PARAFAC 算法和 ESPRIT 算法的估计性能越来越好。由于分集增益，多阵元数可以改进角度估计的性能。同时，比较图 6.2.9 和图 6.2.10 中两个算法的性能曲线可以发现，相同阵元数情况下三线性分解算法的估计性能至少要比 ESPRIT 算法高出 1dB。

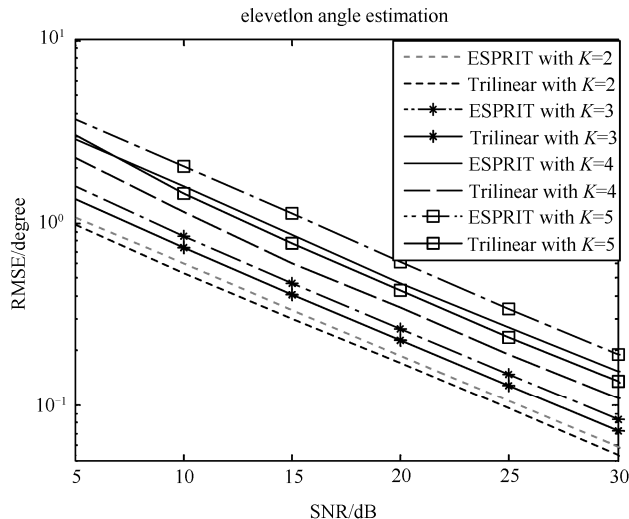


图 6.2.12 不同信源数 (K) 时仰角的估计性能比较 ($M=8$, $L=100$)

图 6.2.11 和图 6.2.12 则分别比较了不同信源数 K 的情况下，该算法和 ESPRIT 算法的仰角和方位角的估计性能 ($L=100$, $M=8$)。从图 6.2.11 和图 6.2.12 可以推断出，随着信源数的增加 PARAFAC 算法和 ESPRIT 算法的估计性能在不断降低，相比较而言，相同信源数条件下 PARAFAC 算法的估计性能要比 ESPRIT 算法要好，而且随着信源数 K 值的不断增加，PARAFAC 算法和 ESPRIT 算法的估计性能之间的差距也越来越大。当 $K=5$ 时，ESPRIT 算法的方位角估计性能已经恶化，而 PARAFAC 算法仍然还具有很好的估计性能。

总的来说，从图 6.2.8~图 6.2.12 可以得出结论：不同参数值情况下 PARAFAC 算法的估计性能一般比 ESPRIT 算法的估计性能至少要高出 0.5dB。这是由于 PARAFAC 算法中在高斯白噪声的情况下能够使三线性分解中的 TALS 方法最优化。

6.2.6 小结

本节研究了基于三线性分解的任意声矢量传感器阵的 DOA 估计，由于三线性分解的唯一性，三线性分解算法被用于对声矢量阵的二维 DOA 估计。将三线性分解方法延用于任意阵提供了未知位置的不规则或随机间隔阵列的盲 DOA 估计。PARAFAC 算法的 DOA 估计性能较之 ESPRIT 算法至少要好 0.5dB；与 MUSIC 算法相比，PARAFAC 算法不需要阵列的几何先验知识或峰值搜索；PARAFAC 算法和 ESPRIT 算法一样，也不需要额外配对。此外，该算法可以支持小样本。

最后, 对两种算法的计算复杂度做一个简略的分析, 总体来说 PARAFAC 算法相较于 ESPRIT 算法有较大的计算复杂度。ESPRIT 算法需要 $O(16M^2L + 64M^3 + K^3)$ 次复数乘法; 然而, PARAFAC 算法中, 每个 TALS 的重复就需要 $O(K^3 + 4MLK)$ 次复数乘法, 而且三线性分解所必需的 TALS 的迭代次数至少是 10 次。

6.3 阵列天线 MC-CDMA 系统中基于平行因子技术的盲多用户检测算法

本节分析阵列天线 MC-CDMA 系统的接收信号具有三线性模型的特征, 提出基于三线性分解的盲多用户检测算法^[10]。该多用户检测算法的性能非常接近于非盲 MMSE 算法, 并且比最小输出能量接收器 (MOE) 和匹配滤波器性能更优。对比非盲算法, 所提算法不需要信道衰落、DOA 信息、导频或者训练信息。而且本节算法完全利用了 MC-CDMA 系统的扩频、空间分集和时间分集序列。

6.3.1 数据模型

假设 MC-CDMA 系统有 K 个用户, 该系统装配有 I 个阵元的均匀线阵接收。第 k 个用户的信息传输结构如图 6.3.1 所示。

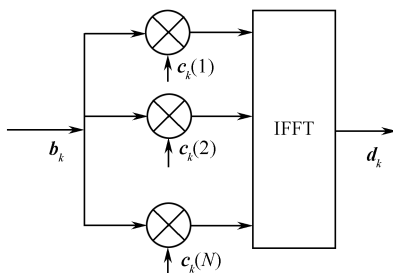


图 6.3.1 第 k 个用户的信息传输结构

第 k 个用户的传输符号序列为 $\mathbf{b}_k = [b_k(1), b_k(2), \dots, b_k(L)]^T$, 其中 $b_k(l) \in \{\pm 1\}$, L 为符号长度, 数据符号 b_k 采用 BPSK 调制, 并且在 N 个子载波上并行传输, 每个子载波分别乘以扩频序列 \mathbf{c}_k , 则 $\mathbf{u}_k = \mathbf{c}_k \mathbf{b}_k^T$, 其中 $\mathbf{c}_k = [c_k(1), c_k(2), \dots, c_k(N)]^T$ 为第 k 个用户的扩频码, 且 $c_k(n) \in \{\pm 1\}$ 。信号 \mathbf{u}_k 通过多载波调制, 多载波调制可用 IFFT 来表示。将多载波调制后的输出信号表示为 $\mathbf{d}_k = \mathbf{F}^H \mathbf{u}_k = \mathbf{F}^H \mathbf{c}_k \mathbf{b}_k^T$, 其中 \mathbf{F} 是 $N \times N$ 的离散傅里叶变换矩阵, \mathbf{F}^H 表示离散傅里叶逆变换。假设 K 个用户单径到达接收天线, 第 i 个天线上的接收信号可以表示为

$$X_i = \sum_{k=1}^K d_k h_{i,k} = \sum_{k=1}^K F^H c_k b_k^T h_{i,k} = F^H C \text{diag}\{h_{i,1}, h_{i,2}, \dots, h_{i,K}\} B^T, \quad i = 1, 2, \dots, I \quad (6.3.1)$$

其中, $h_{i,k}$ 是第 k 个用户在第 i 个天线上的信道响应, $C = [c_1, c_2, \dots, c_K] \in \mathbb{R}^{N \times K}$, $B = [b_1, b_2, \dots, b_K] \in \mathbb{R}^{L \times K}$ 分别代表扩频矩阵和信号矩阵。现在定义信道矩阵为

$$H = \begin{bmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,K} \\ h_{2,1} & h_{2,2} & \dots & h_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ h_{I,1} & h_{I,2} & \dots & h_{I,K} \end{bmatrix} \in \mathbb{C}^{I \times K}, \quad h_{i,k} = h_{j,k} e^{-j2\pi d(i-j)\sin\theta_k/\lambda} \quad (6.3.2)$$

其中 d 为阵元间距, λ 是波长, θ_k 为第 k 个用户的 DOA。式(6.3.2)也可以表示为 $H = A\Phi$, 其中 $\Phi = \text{diag}\{h_{1,1}, h_{1,2}, \dots, h_{1,K}\} \in \mathbb{C}^{K \times K}$, $A = [a(\theta_1), a(\theta_2), \dots, a(\theta_K)] \in \mathbb{C}^{I \times K}$ 分别代表信号衰落对角矩阵和方向矩阵, 且 $a(\theta_k) = [1, e^{-j2\pi d \sin\theta_k/\lambda}, \dots, e^{-j2\pi(I-1)d \sin\theta_k/\lambda}]^T$ 。

式(6.3.1)也可以表示为

$$X_i = S D_i(H) B^T, \quad i = 1, \dots, I \quad (6.3.3)$$

其中 $S = F^H C \in \mathbb{C}^{N \times K}$, $D_i(\cdot)$ 表示取矩阵的第 i 行并由此构造一个对角矩阵。在有噪模型中, 接收信号模型为 $\tilde{X}_i = S D_i(H) B^T + V_i$, $i = 1, 2, \dots, I$, 其中 V_i 为 $N \times L$ 的噪声矩阵。

再次考虑式(6.3.3), 该信号可以表示为三线性模型或者 PARAFAC 模型:

$$x_{n,i,l} = \sum_{k=1}^K s_{n,k} h_{i,k} b_{l,k}, \quad n = 1, \dots, N, \quad i = 1, \dots, I, \quad l = 1, \dots, L \quad (6.3.4)$$

其中 $s_{n,k}$ 表示矩阵 S 的第 (n,k) 元素, $h_{i,k}$ 代表矩阵 H 的第 (i,k) 元素, $b_{l,k}$ 代表矩阵 B 的第 (l,k) 元素。在信号处理领域, PARAFAC 可以被当成 ESPRIT 和联合对角化思想的扩展^[3,4], 式(6.3.4)中的信号可以被当成三线性模型沿着天线方向的切片, 式(6.3.4)中三线性模型的对称性使得矩阵变换成为了可能, 特别地,

$$Y_n = B D_n(S) H^T, \quad n = 1, \dots, N \quad (6.3.5)$$

其中 Y_n 是沿着扩频方向的第 n 个切片。

6.3.2 阵列天线 MC-CDMA 系统中的盲多用户检测算法

1. 三线性分解

TALS 算法是对三线性模型的常用数据检测算法。根据式(6.3.3), 构造矩阵

$$X^{N \times L} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_I \end{bmatrix} = \begin{bmatrix} S D_1(H) \\ S D_2(H) \\ \vdots \\ S D_I(H) \end{bmatrix} B^T = [H \odot S] B^T \quad (6.3.6)$$

其中 \odot 代表 Khatri-Rao 积。最小二乘 (LS) 拟合

$$\min_{S, H, B} \|\tilde{X}^{N \times L} - [H \odot S] B^T\|_F \quad (6.3.7)$$

其中 $\tilde{X}^{N \times L}$ 表示含噪信号。LS 更新矩阵 B 为

$$\mathbf{B}^T = [\mathbf{H} \odot \mathbf{S}]^+ \tilde{\mathbf{X}}^{NI \times L} \quad (6.3.8)$$

类似地, 从第二个方向切片: $\mathbf{Y}_n = \mathbf{B}\mathbf{D}_n(\mathbf{S})\mathbf{H}^T$, $n = 1, 2, \dots, N$, 可以被重写为

$$\mathbf{Y}^{LN \times I} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_N \end{bmatrix} = \begin{bmatrix} \mathbf{B}\mathbf{D}_1(\mathbf{S}) \\ \mathbf{B}\mathbf{D}_2(\mathbf{S}) \\ \vdots \\ \mathbf{B}\mathbf{D}_N(\mathbf{S}) \end{bmatrix} \mathbf{H}^T = [\mathbf{S} \odot \mathbf{B}]\mathbf{H}^T \quad (6.3.9)$$

因此 LS 更新矩阵 \mathbf{H} :

$$\hat{\mathbf{H}}^T = [\mathbf{S} \odot \mathbf{B}]^+ \tilde{\mathbf{Y}}^{LN \times I} \quad (6.3.10)$$

根据式 (6.3.8) 和式 (6.3.10), 矩阵 \mathbf{B} 和 \mathbf{H} 直到条件最小二乘收敛时才停止更新。TALS 算法的优点为易于实现, 保证收敛, 易于扩展到高阶数据, 以及当噪声为加性白高斯时为最佳检测。但是 TALS 算法收敛速度较慢。在本节的算法设计中, 通过采用已知的矩阵 \mathbf{S} 来提高 TALS 算法的收敛速度。

2. 可辨识性

对于式 (6.3.4), $\mathbf{X}_i = \mathbf{S}\mathbf{D}_i(\mathbf{H})\mathbf{B}^T = \mathbf{S}\mathbf{D}_i(\mathbf{A})[\mathbf{B}']^T$, 其中 $\mathbf{B}' = \mathbf{B}\Phi$ 。

定理 6.3.1: $\mathbf{X}_i = \mathbf{S}\mathbf{D}_i(\mathbf{A})[\mathbf{B}']^T$, $i=1, 2, \dots, I$, $\mathbf{S} \in \mathbb{C}^{N \times K}$, $\mathbf{A} \in \mathbb{C}^{I \times K}$, $\mathbf{B}' \in \mathbb{C}^{L \times K}$, 考虑到 \mathbf{A} 具有 Vandermonde 特性, 如果满足

$$k_s + \min(I + k_{B'}, 2K) \geq 2K + 2 \quad (6.3.11)$$

\mathbf{A} 、 \mathbf{B} 和 \mathbf{S} 是唯一的 (存在列模糊和尺度模糊), 也就是说由任何三维矩阵 $\bar{\mathbf{A}}, \bar{\mathbf{B}}, \bar{\mathbf{S}}$ 构造得到的 \mathbf{X}_i ($i = 1, 2, \dots, I$) 可以通过如下的形式得到 \mathbf{A} 、 \mathbf{B} 和 \mathbf{S} :

$$\bar{\mathbf{A}} = \mathbf{A}\Pi\mathbf{A}_1, \quad \bar{\mathbf{B}} = \mathbf{B}'\Pi\mathbf{A}_2 = \mathbf{B}\Phi\Pi\mathbf{A}_2 = \mathbf{B}\Pi\Phi\mathbf{A}_2, \quad \bar{\mathbf{S}} = \mathbf{S}\Pi\mathbf{A}_3 \quad (6.3.12)$$

其中 Π 为列模糊矩阵, $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$ 为尺度模糊矩阵, 并且满足 $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3 = \mathbf{I}_K$, 其中 \mathbf{I}_K 为 $K \times K$ 单位矩阵。

矩阵 \mathbf{S} 已知可以解决列尺度模糊问题, 注意到式 (6.3.12) 可变为

$$\bar{\mathbf{A}} = \mathbf{A}\mathbf{A}_1, \quad \bar{\mathbf{B}} = \mathbf{B}\Phi\mathbf{A}_2 \quad (6.3.13)$$

$\mathbf{A}_1, \mathbf{A}_2$ 为对角尺度模糊矩阵且满足 $\mathbf{A}_1\mathbf{A}_2 = \mathbf{I}_K$ 。

其中存在尺度模糊问题, 其解决方法为自动增益控制、正交化和不同的编码/解码方法或者嵌入信息。

对于信源独立的信号 $k_{B'} = \min(L, K)$ 和正交扩频矩阵 $k_s = \min(N, K)$, 式 (6.3.11) 可变为

$$\min(N, K) + \min(I + \min(L, K), 2K) \geq 2K + 2 \quad (6.3.14)$$

实际中, $N \geq K$ 且 $\min(N, K) = K$, 因此

$$\min(I + \min(L, K), 2K) \geq K + 2 \quad (6.3.15)$$

当 $L \geq K \geq 2$ 时, 注意到 $\min(L, K) = K$, 此时的辨识条件为 $I \geq 2$ 。当 $L \leq K, K \geq 2$ 时, 辨识条件为 $L + I \geq K + 2$, 在这种情况下, 本节算法支持小样本。

3. 盲多用户检测的步骤

到此为止, 已经提出了基于三线性分解的 MC-CDMA 系统多用户检测算法 (Trilinear-MUDE), 主要步骤总结如下。

步骤 1: 随机初始化信源矩阵 \mathbf{B} 和信道矩阵 \mathbf{H} 。

步骤 2: 根据式 (6.3.8) 利用 LS 更新矩阵 \mathbf{B} 。

步骤 3: 根据式 (6.3.10) 利用 LS 更新矩阵 \mathbf{H} 。

步骤 4: 重复步骤 2 到步骤 3 直到收敛。

步骤 5: 利用信源矩阵 \mathbf{B} 实现盲多用户检测。

4. 复杂度分析

非盲的 MMSE 算法可由式 (6.3.7) 得到, 即

$$\mathbf{B}_{\text{mmse}}^T = [\mathbf{A}^H \mathbf{A} + \mathbf{I}_{NI} \frac{1}{\text{SNR}}]^{-1} \tilde{\mathbf{X}}^{NI \times L} \quad (6.3.16)$$

其中 $\mathbf{A} = [\mathbf{H} \odot \mathbf{S}]$, \mathbf{I}_{NI} 是 $NI \times NI$ 的单位矩阵, 信噪比 (SNR) 将在式 (6.3.17) 中定义。

对比非盲的 MMSE 算法, 本节算法的计算复杂度更高。非盲的 MMSE 算法复杂度为 $O(NIK^2 + K^3 + NILK + K^2L)$, 本节算法每次迭代的复杂度为 $O(NIK^2 + NLK^2 + 2K^3 + 2NILK + K^2L + K^2I)$ 。图 6.3.2 中显示了 PARAFAC 分解大概需要 8 次迭代。

6.3.3 仿真结果

考虑接收的带噪信号 $\tilde{\mathbf{X}}_i = \mathbf{SD}_i(\mathbf{H})\mathbf{B}^T + \mathbf{V}_i$, $i = 1, 2, \dots, I$, 定义 SNR:

$$\text{SNR} = 10 \log_{10} \frac{\sum_{i=1}^I \|\mathbf{SD}_i(\mathbf{H})\mathbf{B}^T\|_F^2}{I \times N \times \sum_{i=1}^I \|\mathbf{V}_i\|_F^2} \text{ dB} \quad (6.3.17)$$

我们使用 Montel Carlo 仿真来验证 Trilinear-MUDE 算法的性能, 其中 Montel Carlo 仿真的次数为 1000, 均匀线阵的接收天线数为 8。假设每个用户的信道都独立, 采用 BPSK 调制, 噪声为加性高斯白噪声 (AWGN), 子载波数目为 32, L 为快拍数, K 为用户数。

首先检测算法的收敛性能。三线性拟合中残差平方和 (SSR) = $\sum_{n=1}^N \sum_{i=1}^I \sum_{l=1}^L [\tilde{x}_{n,i,l} - \sum_{k=1}^K \hat{s}_{n,k} \hat{h}_{i,k} \hat{b}_{l,k}]^2$, 其中 $\tilde{x}_{n,i,l}$ 为含噪信号。定义 DSSR = $\text{SSR}_i - \text{SSR}_0$, 其中 SSR_i 为 SSR 的第 i 次迭代, SSR_0 为精确的 SSR。图 6.3.2 显示了在 $K = 3$ 、 $L = 50$ 和 $\text{SNR} = 16\text{dB}$ 下的算法收敛性能。由图 6.3.2 可以发现, 算法收敛性是优秀的。

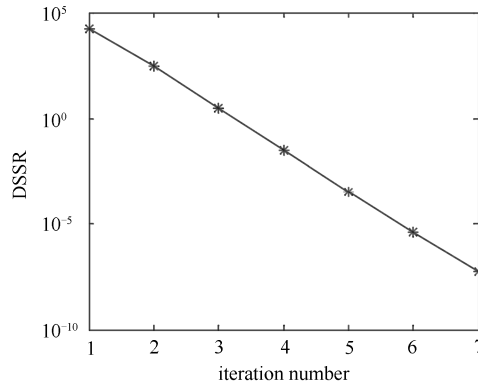


图 6.3.2 算法收敛性能

我们将本节算法和非盲最小均方误差(MMSE)接收器、盲最小输出能量接收器(MOE)以及匹配滤波器进行了对比, MMSE 接收器提供了盲算法的性能边界。对比 Trilinear-MUDE 算法, 盲 MMSE 接收器需要精确的 SNR、DOA 和信道衰落信息, Trilinear-MUDE 算法则不需要这些信息。

这些算法在不同 L 下的多用户检测性能如图 6.3.3~图 6.3.6 所示。其中用户数都为 4, DOA 分别为 10° 、 20° 、 30° 和 40° 。

图 6.3.3 表示 $L=50$ 时的仿真结果。从图 6.3.3 可知, 本节算法性能很接近非盲 MMSE 算法, 并且明显优于 MOE 接收器算法和匹配滤波器, Trilinear-MUDE 算法和 MMSE 算法拥有相同的算法性能。图 6.3.4 和图 6.3.5 分别显示了 $L=20$ 和 $L=10$ 时仿真结果, 可以看出本节算法的性能在图 6.3.4 中比盲 MMSE 接收器算法性能少 0.2dB, 在图 6.3.5 中大概少 0.5dB。由图 6.3.3~图 6.3.5 可以得出, 本节算法性能和非盲 MMSE 性能差距随着 L 的减少而增加。图 6.3.6 显示在特殊小样本 $L=5$ 时的性能, 本节算法在小样本下依旧性能良好。

图 6.3.7 显示了 Trilinear-MUDE 在不同 K 下的算法性能。当 MC-CDMA 系统中的用户数变化时, 多用户检测算法的性能也不同。用户数分别为 3、4 和 5。很明显, 随着用户数的减少, 算法性能上升。

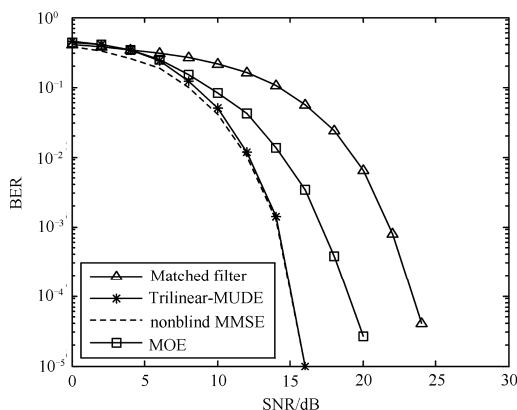


图 6.3.3 $L=50$ 时的算法性能

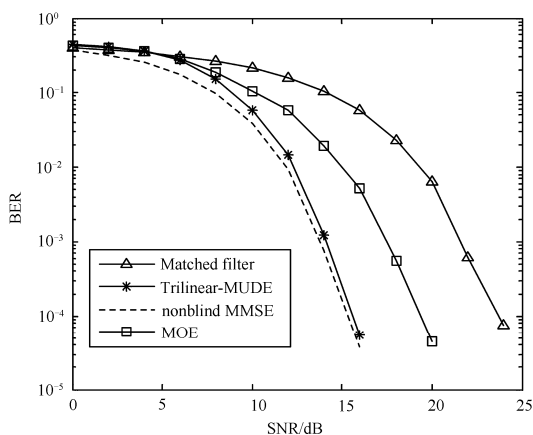


图 6.3.4 $L=20$ 时的算法性能对比

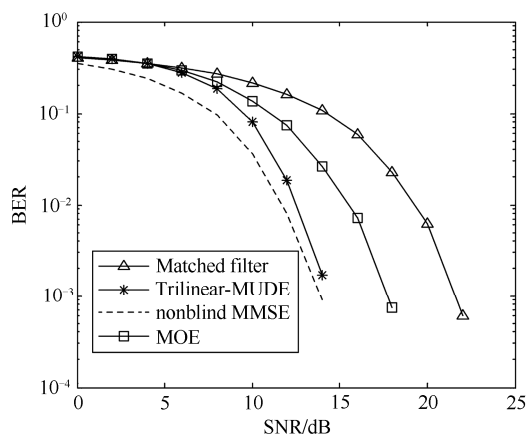


图 6.3.5 $L = 10$ 时的算法性能对比

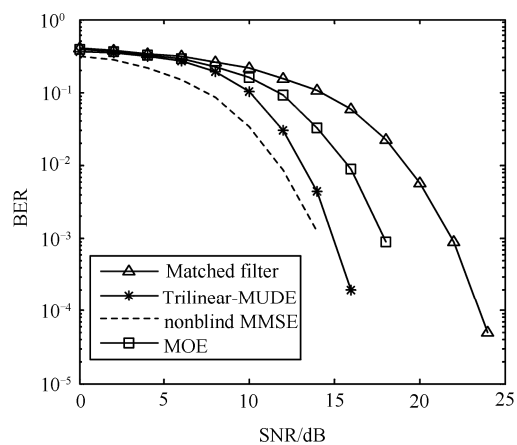


图 6.3.6 $L = 5$ 时的算法性能对比

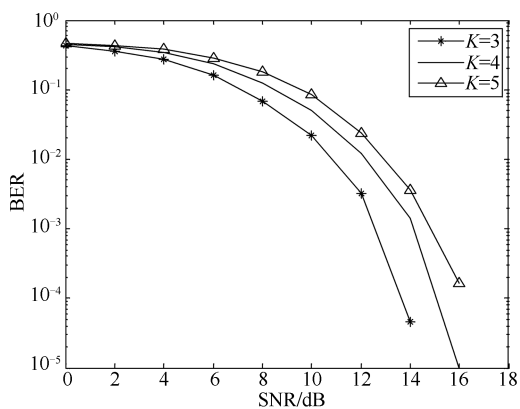


图 6.3.7 不同 K 下的算法性能

6.4 单基地 MIMO 雷达中基于自适应 PARAFAC-RLST 的 DOA 跟踪算法

本节将自适应 PARAFAC^[12]应用于 MIMO 雷达系统中, 实现单基地 MIMO 雷达角度跟踪^[13,14]。

6.4.1 数据模型

在单基地 MIMO 雷达系统中, 发射阵列和接收阵列都采用简单的均匀阵列, 发射阵列上的阵元数为 M , 接收阵列上的阵元数为 N 。阵元间距取半波长, 即 $d = \lambda_s / 2$, λ_s 是信号波长。假设空间有 K 个互不相关的目标处于高斯白噪声下, 且以恒定地速度缓慢运动。则在 t 时刻, 接收端匹配滤波器的输出可以表示为

$$\begin{aligned} \mathbf{r}(t) &= [r_1(t), r_2(t), \dots, r_{MN}(t)]^T = (\mathbf{A}_R(\theta(t)) \odot \mathbf{A}_T(\theta(t)))\mathbf{s}(t) + \mathbf{n}(t) \\ &= \mathbf{A}(\theta(t))\mathbf{s}(t) + \mathbf{n}(t) \end{aligned} \quad (6.4.1)$$

其中, $\theta(t) = [\theta_1(t), \theta_2(t), \dots, \theta_K(t)]$, $\mathbf{A}_T(\theta(t))$ 和 $\mathbf{A}_R(\theta(t))$ 分别是在 t 时刻, 发射阵和接收阵的导向矩阵。 $\mathbf{A}(\theta(t)) = \mathbf{A}_R(\theta(t)) \odot \mathbf{A}_T(\theta(t)) = [\mathbf{a}_r(\theta_1(t)) \otimes \mathbf{a}_t(\theta_1(t)), \dots, \mathbf{a}_r(\theta_K(t)) \otimes \mathbf{a}_t(\theta_K(t))]$ 。矩阵 $\mathbf{A}_T(\theta(t))$ 的第 i 列向量 $\mathbf{a}_i(t) = [1, \exp(-j\pi \sin \theta_i(t)), \dots, \exp(-j(M-1)\pi \sin \theta_i(t))]^T$ 。 $\mathbf{s}(t) \in \mathbb{C}^{K \times 1}$, $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_K(t)]^T$, $s_k(t) = \beta_k(t)e^{j2\pi f_k t}$ 。 $\mathbf{n}(t) \in \mathbb{C}^{MN \times 1}$ 是均值为 $\mathbf{0}$, 方差为 $\sigma^2 \mathbf{I}_{MN}$ 的高斯白噪声。

6.4.2 利用自适应 PARAFAC-RLST 进行 DOA 跟踪

三阶张量 $\chi \in \mathbb{C}^{M \times J \times N}$ 的 PARAFAC 分解可以表示为

$$\chi = \sum_{k=1}^K \mathbf{a}_k \circ \mathbf{b}_k \circ \mathbf{c}_k \quad (6.4.2)$$

其中, \mathbf{a}_k 、 \mathbf{b}_k 、 \mathbf{c}_k 分别表示为加载矩阵 $\mathbf{A}_T \in \mathbb{C}^{M \times K}$ 、 $\mathbf{S} \in \mathbb{C}^{J \times K}$ 、 $\mathbf{A}_R \in \mathbb{C}^{N \times K}$ 的第 k 列。将单基地 MIMO 雷达的角度估计与三线性模型联系起来, J 为快拍数, \circ 表示向量的外积。

根据其对称性, 可以得到以下矩阵:

$$\mathbf{R}^{(1)} = (\mathbf{A}_R \odot \mathbf{A}_T) \mathbf{S}^T, \quad \mathbf{R}^{(2)} = (\mathbf{S} \odot \mathbf{A}_R) \mathbf{A}_T^T, \quad \mathbf{R}^{(3)} = (\mathbf{A}_T \odot \mathbf{S}) \mathbf{A}_R^T \quad (6.4.3)$$

考虑三阶张量 $\chi(t) \in \mathbb{C}^{M \times J(t) \times N}$ 的 PARAFAC 的估计值, 在时刻 t 有

$$\mathbf{R}^{(1)}(t) \approx \mathbf{A}(t) \mathbf{S}^T(t) \quad (6.4.4)$$

在时刻 $t+1$, 可以得到

$$\mathbf{R}^{(1)}(t+1) = [\mathbf{R}^{(1)}(t), \mathbf{r}(t+1)] \quad (6.4.5)$$

其中, $\mathbf{r}(t+1)$ 是附加在 $\mathbf{R}^{(1)}(t)$ 后的一个新的切片。 $\mathbf{A}(t) = \mathbf{A}_R(t) \odot \mathbf{A}_T(t)$, $\mathbf{A}_R(t)$ 和 $\mathbf{A}_T(t)$ 分别表示单基地 MIMO 雷达中接收阵和发射阵的导向矩阵。

假设矩阵 \mathbf{A} 在时刻 t 和时刻 $t+1$ 之间变化缓慢, 则有 $\mathbf{A}(t+1) \approx \mathbf{A}(t)$, 则 $\mathbf{S}^T(t+1) = [\mathbf{S}^T(t), \mathbf{s}^T(t+1)]^T$, 其中 $\mathbf{s}^T(t+1) = \mathbf{A}^+(t)\mathbf{r}(t+1)$ 。 $\mathbf{A}(t+1)$ 的最小二乘更新可以表示成 $\mathbf{A}(t+1) = \mathbf{R}^{(1)}(t+1)(\mathbf{S}^T(t+1))^+$ 。

给定 $\mathbf{A}(t+1)$, 通过 $\mathbf{A}(t+1)$ 取代 $\mathbf{A}(t)$ 获得 $\mathbf{r}(t+1)$ 的一个更新。最后从矩阵 $\mathbf{A}(t+1)$ 得到 $\mathbf{A}_R(t+1)$ 和 $\mathbf{A}_T(t+1)$ 。

对于一个窗长度为 W ($W > K$) 的截断窗 (Truncated Window), 可以定义一个加权矩阵

$$\mathbf{R}_{TW}^{(1)}(t+1) = [\mathbf{r}(t+2-W), \dots, \mathbf{r}(t), \mathbf{r}(t+1)] \quad (6.4.6)$$

定义 $\chi(t+1) \in \mathbb{C}^{M \times W \times N}$ 表示 W 个最新切片构成的张量, $\chi(t+1)$ 的 PARAFAC 分解的估计可以通过使下面截断窗最小二乘准则最小化得到:

$$\min_{\{\mathbf{A}(t+1), \mathbf{S}_{TW}(t+1)\}} (\phi^{TW}(t+1))$$

其中, $\phi^{TW}(t+1) = \sum_{\tau=1}^N \lambda^{W-\tau} \|\mathbf{r}(\mu+\tau) - \mathbf{A}(t+1)\mathbf{s}^T(\mu+\tau)\|^2$, λ 是遗忘因子, $\mu = t+1-W$ 和 $\mathbf{S}_{TW}(t+1)$ 由 $\mathbf{S}(t+1)$ 的最后 W 行构成。

然后可由 $\mathbf{R}_{TW}^{(1)}(t+1)$ 得到加权观察矩阵 $\mathbf{R}_{TW}(t+1)$, 即

$$\mathbf{R}_{TW}(t+1) = \mathbf{R}_{TW}^{(1)}(t+1)\mathbf{A} \quad (6.4.7)$$

其中, $\mathbf{A} = \text{diag}([\lambda^{W-1/2}, \lambda^{W-2/2}, \dots, \lambda^{1/2}, 1])$ 。

$\mathbf{R}_{TW}^{(1)}(t)$ 的更新规则是

$$[\lambda^{W/2}\mathbf{r}(t+1-W), \mathbf{R}_{TW}(t+1)] = [\lambda^{1/2}\mathbf{R}_{TW}(t), \mathbf{r}(t+1)] \quad (6.4.8)$$

PARAFAC-RLST 是一种自适应跟踪算法, 基于最小化的加权最小二乘标准。给定 $\mathbf{s}(t+1)$ 的一个初始估计, 可以对 $\mathbf{A}(t+1)$ 进行递归更新。相对于 $\mathbf{A}(t+1)$, $\nabla\phi^{TW}(t+1) \in \mathbb{C}^{MN \times K}$ 是 $\phi^{TW}(t+1)$ 的梯度。有

$$\nabla\phi^{TW}(t+1) = 2 \sum_{\tau=1}^W \lambda^{W-\tau} (\mathbf{r}(\mu+\tau) - \mathbf{A}(t+1)\mathbf{s}^T(\mu+\tau))\mathbf{s}^*(\mu+\tau) \quad (6.4.9)$$

令 $\nabla\phi^{TW}(t+1) = 0$, 可以得到

$$\mathbf{A}(t+1) = \mathbf{D}_{TW}(t+1)\mathbf{P}_{TW}^{-1}(t+1) \quad (6.4.10)$$

其中, $\mathbf{D}_{TW}(t+1) = \sum_{\tau=1}^W \lambda^{W-\tau} \mathbf{r}(\mu+\tau)\mathbf{s}^*(\mu+\tau)$, $\mathbf{P}_{TW}(t+1) = \sum_{\tau=1}^W \lambda^{W-\tau} \mathbf{s}^T(\mu+\tau)\mathbf{s}^*(\mu+\tau)$ 。

通过递归更新有

$$\mathbf{D}_{TW}(t+1) = \lambda\mathbf{D}_{TW}(t) + \mathbf{r}(t+1)\mathbf{s}^*(t+1) - \lambda^W \mathbf{r}(\mu)\mathbf{s}^*(\mu) \quad (6.4.11)$$

$$\mathbf{P}_{TW}(t+1) = \lambda\mathbf{P}_{TW}(t) + \mathbf{s}^T(t+1)\mathbf{s}^*(t+1) - \lambda^W \mathbf{s}^T(\mu)\mathbf{s}^*(\mu) \quad (6.4.12)$$

假设 $\mathbf{Q}_{TW} = \mathbf{P}_{TW}^{-1}$, 两次引用矩阵求逆引理得

$$\mathbf{Q}_{TW}(t+1) = \tilde{\mathbf{Q}} + \frac{\lambda^W \tilde{\mathbf{Q}} \mathbf{s}^T(\mu) \mathbf{s}^*(\mu) \tilde{\mathbf{Q}}}{1 - \lambda^W \mathbf{s}^*(\mu) \tilde{\mathbf{Q}} \mathbf{s}^T(\mu)} \quad (6.4.13)$$

其中, $\tilde{\mathbf{Q}} = \lambda^{-1} \mathbf{Q}_{TW}(t) - \frac{\lambda^{-2} \mathbf{Q}_{TW}(t) \mathbf{s}^T(t+1) \mathbf{s}^*(t+1) \mathbf{Q}_{TW}(t)}{1 + \lambda^{-1} \mathbf{s}^*(t+1) \mathbf{Q}_{TW}(t) \mathbf{s}^T(t+1)}$ 。

最后由 $\mathbf{D}_{TW}(t+1)$ 和 $\mathbf{P}_{TW}^{-1}(t+1)$ 对 $\mathbf{A}(t+1)$ 进行递归更新, 有

$$\mathbf{A}(t+1) = \mathbf{D}_{TW}(t+1) \mathbf{P}_{TW}^{-1}(t+1) \quad (6.4.14)$$

现在假设给定 $\mathbf{A}(t+1)$ 的先前估计, 想要对 $\mathbf{S}(t+1)$ 的最后一行 $\mathbf{s}(t+1)$ 进行估计, 由于 $\mathbf{A}^+(t+1) = \mathbf{P}(t+1) \mathbf{D}^+(t+1)$, $\mathbf{s}(t+1)$ 的最小二乘估计可以表示为

$$\mathbf{s}^T(t+1) = \mathbf{P}_{TW}^+(t+1) \mathbf{D}_{TW}^+(t+1) \mathbf{r}(t+1) \quad (6.4.15)$$

算法的主要目的是通过 $\mathbf{D}_{TW}^+(t+1)$ 、 $\mathbf{P}_{TW}^{-1}(t+1)$ 的更新方程分别是:

$$\mathbf{D}_{TW}^+(t+1) = (\lambda \mathbf{D}_{TW}(t) + \mathbf{r}(t+1) \mathbf{s}^*(t+1) - \lambda^W \mathbf{r}(t) \mathbf{s}^*(t))^+ \quad (6.4.16)$$

$$\mathbf{P}_{TW}^{-1}(t+1) = (\lambda \mathbf{P}_{TW}(t) + \mathbf{s}^T(t+1) \mathbf{s}^*(t+1) - \lambda^W \mathbf{s}^T(t) \mathbf{s}^*(t))^{-1} \quad (6.4.17)$$

获取它们的递归更新值。

至此, 从以上步骤可以得到 $\mathbf{A}_R(t+1)$ 或 $\mathbf{A}_T(t+1)$ 的估计值 $\hat{\mathbf{A}}_R(t+1)$ 或 $\hat{\mathbf{A}}_T(t+1)$, 然后便可利用矩阵的旋转不变性求得 $t+1$ 时刻各个目标的角度, 分别为 $\theta_1(t+1), \theta_2(t+1), \dots, \theta_K(t+1)$ 。但是求出来的角度并没有和目标进行关联, 必须采用关联技术才能对目标进行准确跟踪。

现有的数据关联算法有最近邻法、航迹分裂法、0-1 整数规划法等, 文献[15]利用不同时刻观测的估计值和目标位置关系来将不同目标与其相对应的 DOA 进行关联。文献[16]考虑的是用两个均匀阵列分别对空间目标进行两次估计, 然后通过重构协方差矩阵实现角度关联。本节可采用文献[16]角度关联的思想或其他关联技术来对单基地 MIMO 雷达中的 DOA 进行关联。

假设 t 时刻各个目标的角度是已知的或已由其他算法估计出, 分别为 $\theta_1(t), \theta_2(t), \dots, \theta_K(t)$ 。构建信号协方差矩阵 $\hat{\Gamma}$, $\hat{\Gamma} = (\hat{\mathbf{A}}_R(t))^+ \hat{\mathbf{A}}_R(t) \mathbf{S} \hat{\mathbf{A}}_R^H(t+1) [(\hat{\mathbf{A}}_R(t+1))^+]^H$, 找出矩阵 $\hat{\Gamma}$ 中绝对值最大元素的位置 (j_1, k_1) ; 然后去掉该值所在的行和列, 找出剩下元素中绝对值最大的元素位置 (j_2, k_2) , 以此类推, 直到找出第 K 个元素位置。根据各个最大元素的位置信息, 可将角度的估计值与其相对应的目标关联起来。

至此, 单基地 MIMO 雷达中基于自适应 PARAFAC-RLST 的 DOA 跟踪算法已给出, 主要步骤如下:

① 首先为 \mathbf{S}^T 创建 $\mathbf{s}^T(t+1)$ 的一个初始估计。

② 根据式 (6.4.11)、式 (6.4.12) 和式 (6.4.17), 通过两次引用求逆引理对 $\mathbf{D}, \mathbf{P}, \mathbf{D}^+, \mathbf{P}^{-1}$ 进行递归更新。

③ 由式 (6.4.15) 对 $\mathbf{s}^T(t+1)$ 重新估计, 并获取 $\mathbf{S}^T(t+1)$ 。

④ 更新矩阵 $\mathbf{A}(t+1)$, 然后可由 $\mathbf{A}(t+1)$ 得到 $\mathbf{A}_R(t+1)$ 和 $\mathbf{A}_T(t+1)$ 的估计值。

⑤ 然后根据 $\mathbf{A}_R(t+1)$ 或 $\mathbf{A}_T(t+1)$ 的估计值便可求得该时刻对应的 DOA。

⑥ 最后对估计出的 DOA 进行关联运算。

⑦ 再重复步骤 1~6, 对下个时刻的角度进行跟踪估计。

6.4.3 复杂度分析

相比于批处理 (batch) PARAFAC 算法, PARAFAC-RLST 算法的复杂度低, 它避免了 batch-PARAFAC 算法在 $t+1$ 时刻的三阶张量分解而三阶张量分解的计算复杂度很高。本节 PARAFAC-RLST 算法的复杂度为 $K^2(16MN + 72) + K(144MN + 10N + 20)$ 。而 batch-PARAFAC 算法的复杂度为 $88K^3 + 64K^2(WN + MN + MW) + 8K(3MNW + WN + MN + MW)$ 。

6.4.4 仿真结果

定义均方根误差 RMSE 为

$$\text{RMSE} = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{Q} \sum_{q=1}^Q \left[\frac{1}{T} \sum_{t=1}^T (\hat{\theta}_{k,q,t} - \theta_{k,q,t})^2 \right]} \quad (6.4.18)$$

其中, $\hat{\theta}_{k,q,t}$ 是第 k 个目标的 DOA θ_k 在时刻 t 的第 q 次 Monte Carlo 试验的估计值。

M 、 N 、 J 、 K 和 Q 分别表示发射阵元数、接收阵元数、快拍数、目标数和 Monte Carlo 试验次数。截断窗的长度是 W , $W=10$ 。Monte Carlo 试验次数 $Q=1000$ 。除了图 6.4.5~图 6.4.7, 其他图中取 $M=8$ 、 $N=6$ 、 $J=100$ 和 $K=3$, 遗忘因子 λ 取 0.97。在 DOA 跟踪过程中, 观察的切片数 T 为 200, 采样间隔是 1s。

仿真 1: 图 6.4.1~图 6.4.3 显示了 SNR=5dB 下本节算法在不同 DOA 变化轨迹下的跟踪结果。从图 6.4.1~图 6.4.3 可以看出, 本节算法不仅可以有效地对单基地 MIMO 雷达中的 DOA 进行跟踪, 而且可以有效地工作在 DOA 变化轨迹相交叉的情况下。

仿真 2: 该仿真以图 6.4.1 所示的 DOA 变化轨迹来分析的, 以下仿真的参数与此相同。图 6.4.4 显示了本节算法和 batch-PARAFAC 算法的 DOA 跟踪性能比较, 可以看出本节算法的 DOA 跟踪性能接近复杂度高的 batch-PARAFAC 算法, 但是本节算法的复杂度远低于 batch-PARAFAC 算法。

仿真 3: 图 6.4.5~图 6.4.7 分别显示了本节算法在不同发射阵元数 M 下、不同接收阵元数 N 下和不同目标数 K 下的 DOA 跟踪性能比较。图 6.4.5 中 $N=6$ 、 $J=100$ 和 $K=3$; 图 6.4.6 中 $M=8$ 、 $J=100$ 和 $K=3$; 图 6.4.7 中 $M=8$ 、 $N=6$ 和 $J=100$ 。从图 6.4.5 和图 6.4.6 可以看出, 随着阵元数 M/N 的增加, 本节算法的 DOA 跟踪性能变好; 从图 6.4.7 可以看出, 空间的目标数 K 减少, 本节算法的 DOA 跟踪性能变好, 说明目标数减小时, 目标间的干扰也减小。

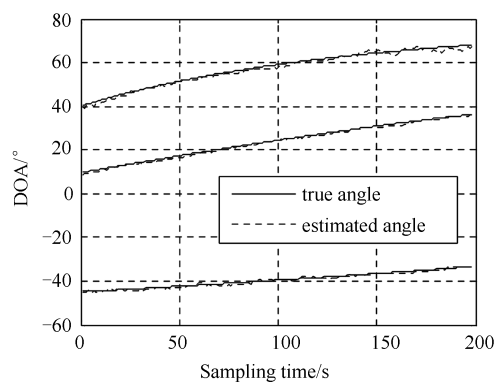


图 6.4.1 SNR=5dB 下本节算法的 DOA 跟踪性能

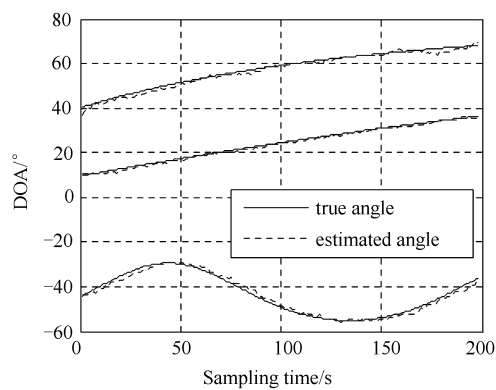


图 6.4.2 SNR=5dB 下本节算法的 DOA 跟踪性能

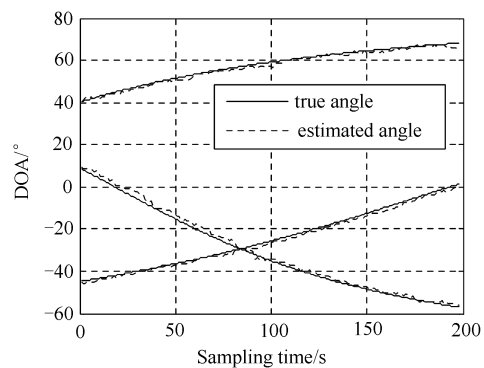


图 6.4.3 SNR=5dB 下本节算法的 DOA 跟踪性能

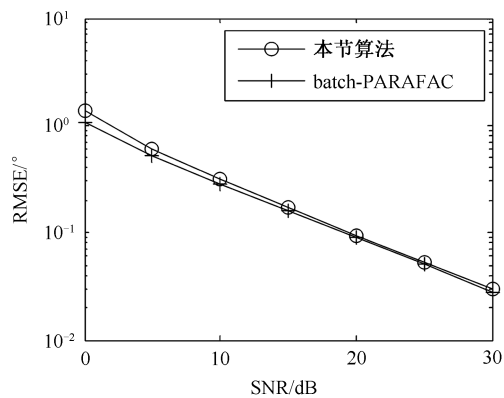


图 6.4.4 不同算法下 DOA 跟踪性能比较

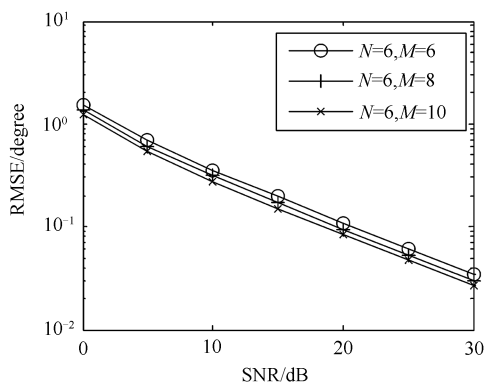


图 6.4.5 不同发射阵元数 M 下的 DOA 跟踪性能

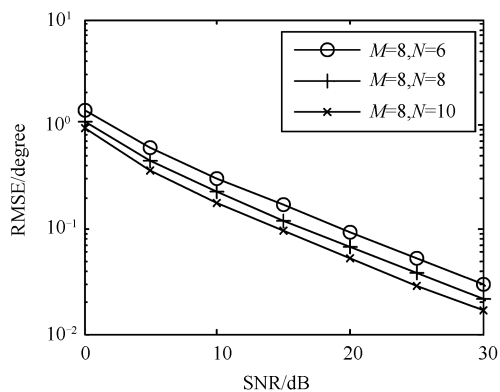
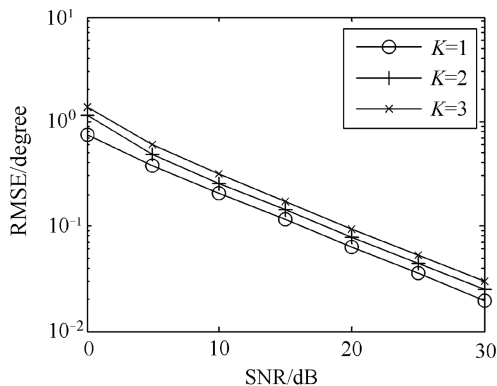


图 6.4.6 不同接收阵元数 N 下的 DOA 跟踪性能

图 6.4.7 不同目标数 K 下的 DOA 跟踪性能

6.5 基于非圆 PARAFAC 任意声矢量阵列下 2D-DOA 估计

本节提出一种非圆 (Non-Circular, NC) -PARAFAC 算法, 该算法可以用于任意声矢量阵列下非圆信号源的 2D-DOA 估计^[17]。相比 PARAFAC 以及传播算子 (Propagator Method, PM) 算法, 本节算法具有更好的角度估计性能, 而且该算法的计算复杂度低于 PARAFAC 算法。本节算法利用信号的非圆特性来提升角度估计的性能, 且适用于在任意声矢量传感器阵且阵元位置分布未知情况下进行角度估计。另外, 本节算法可以实现 2D-DOA 估计值的自动配对。同时, 本节还将分析算法的复杂度, 并推导非圆信号源 DOA 估计的 CRB。

6.5.1 数据模型

考虑 K 个平行波照射在一个声矢量阵列上, 该阵列包含 M 个在三维空间中未知位置的阵元, 如图 6.5.1 所示。假设信号为远场非圆信号, 即信源足够远, 使得来波近似为平行波, 同时假设噪声为加性且服从均值为 0、方差为 σ^2 的高斯分布。第 k 个信号的波达方向为 (ϕ_k, φ_k) , 其中 ϕ_k 和 φ_k 分别表示方位角和仰角。令 $\theta_k = [\phi_k, \varphi_k]^T$, 即第 k 个信源的 2D-DOA。

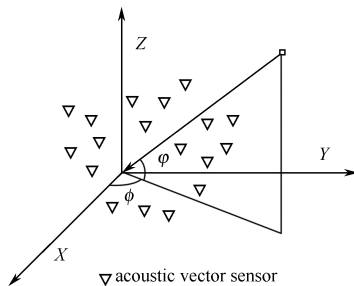


图 6.5.1 声矢量传感器阵列几何结构

根据文献[18, 19], 包含 M 个声矢量传感器阵列输出为

$$\begin{cases} \mathbf{p}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}_p(t) \\ \mathbf{x}(t) = \mathbf{A}\Phi_x\mathbf{s}(t) + \mathbf{n}_x(t) \\ \mathbf{y}(t) = \mathbf{A}\Phi_y\mathbf{s}(t) + \mathbf{n}_y(t) \\ \mathbf{z}(t) = \mathbf{A}\Phi_z\mathbf{s}(t) + \mathbf{n}_z(t) \end{cases} \quad (6.5.1)$$

其中, $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_K]$, $\mathbf{a}_k = [q_1(\theta_k), q_2(\theta_k), \dots, q_M(\theta_k)]^T$, $q_m(\theta_k) = e^{j2\pi\tau_{mk}}$ 。 τ_{mk} 是第 k 个信号在第 m 个阵元与参考点之间的时延, $\mathbf{s}(t)$ 包含 K 个信号。 $\mathbf{n}_p(t)$ 、 $\mathbf{n}_x(t)$ 、 $\mathbf{n}_y(t)$ 和 $\mathbf{n}_z(t)$ 是声矢量阵列的接收噪声。 Φ_x 、 Φ_y 和 Φ_z 为

$$\Phi_x = \text{diag}(\cos\phi_1 \cos\varphi_1, \cos\phi_2 \cos\varphi_2, \dots, \cos\phi_K \cos\varphi_K) \quad (6.5.2)$$

$$\Phi_y = \text{diag}(\sin\phi_1 \cos\varphi_1, \sin\phi_2 \cos\varphi_2, \dots, \sin\phi_K \cos\varphi_K) \quad (6.5.3)$$

$$\Phi_z = \text{diag}(\sin\varphi_1, \sin\varphi_2, \dots, \sin\varphi_K) \quad (6.5.4)$$

考虑信道状态信息在 L 个发射符号之间是恒定的, 有

$$\begin{cases} \mathbf{P} = [\mathbf{p}(1), \mathbf{p}(2), \dots, \mathbf{p}(L)] \\ \mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(L)] \\ \mathbf{Y} = [\mathbf{y}(1), \mathbf{y}(2), \dots, \mathbf{y}(L)] \\ \mathbf{Z} = [\mathbf{z}(1), \mathbf{z}(2), \dots, \mathbf{z}(L)] \end{cases} \quad (6.5.5)$$

构造下面的矩阵 $\mathbf{X}_E \in \mathbb{C}^{4M \times L}$:

$$\mathbf{X}_E = \begin{bmatrix} \mathbf{P} \\ \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix} = \begin{bmatrix} \mathbf{A} \\ \mathbf{A}\Phi_x \\ \mathbf{A}\Phi_y \\ \mathbf{A}\Phi_z \end{bmatrix} \mathbf{S} + \begin{bmatrix} \mathbf{N}_p \\ \mathbf{N}_x \\ \mathbf{N}_y \\ \mathbf{N}_z \end{bmatrix} \quad (6.5.6)$$

其中, $\mathbf{S} = [\mathbf{s}(1), \mathbf{s}(2), \dots, \mathbf{s}(L)] \in \mathbb{C}^{K \times L}$ 是包含 L 个快拍的信号矩阵, $\mathbf{N}_p = [\mathbf{n}_p(1), \mathbf{n}_p(2), \dots, \mathbf{n}_p(L)]$, $\mathbf{N}_x = [\mathbf{n}_x(1), \mathbf{n}_x(2), \dots, \mathbf{n}_x(L)]$, $\mathbf{N}_y = [\mathbf{n}_y(1), \mathbf{n}_y(2), \dots, \mathbf{n}_y(L)]$, $\mathbf{N}_z = [\mathbf{n}_z(1), \mathbf{n}_z(2), \dots, \mathbf{n}_z(L)]$ 为噪声矩阵。根据非圆信号特性, 信号源矩阵可以表示为

$$\mathbf{S} = \boldsymbol{\psi} \mathbf{S}_0 \quad (6.5.7)$$

其中 $\boldsymbol{\psi} = \text{diag}\{e^{i\psi_1}, \dots, e^{i\psi_K}\}$, $e^{i\psi_p} \neq e^{i\psi_q}$ 即 $p \neq q$, $\mathbf{S}_0 \in \mathbb{R}^{K \times L}$, $\mathbf{S}_0 = \mathbf{S}_0^*$ 。

6.5.2 基于 NC-PARAFAC 的 2D-DOA 估计算法

1. 数据重构

扩展 \mathbf{X}_E , 数据矩阵可以被重构为 $\mathbf{Y} \in \mathbb{R}^{8M \times L}$

$$\mathbf{Y} = \begin{bmatrix} \text{Re}\{\mathbf{X}_E\} \\ \text{Im}\{\mathbf{X}_E\} \end{bmatrix} \quad (6.5.8)$$

通过行交换, 得到 \mathbf{V} :

$$V = \begin{bmatrix} \text{Re}\{\mathbf{A}\mathbf{S} + \mathbf{N}_1\} \\ \text{Im}\{\mathbf{A}\mathbf{S} + \mathbf{N}_1\} \\ \text{Re}\{\mathbf{A}\Phi_x\mathbf{S} + \mathbf{N}_2\} \\ \text{Im}\{\mathbf{A}\Phi_x\mathbf{S} + \mathbf{N}_2\} \\ \text{Re}\{\mathbf{A}\Phi_y\mathbf{S} + \mathbf{N}_3\} \\ \text{Im}\{\mathbf{A}\Phi_y\mathbf{S} + \mathbf{N}_3\} \\ \text{Re}\{\mathbf{A}\Phi_z\mathbf{S} + \mathbf{N}_4\} \\ \text{Im}\{\mathbf{A}\Phi_z\mathbf{S} + \mathbf{N}_4\} \end{bmatrix} = \begin{bmatrix} \text{Re}\{\mathbf{A}\mathbf{S}\} + \text{Re}\{\mathbf{N}_p\} \\ \text{Im}\{\mathbf{A}\mathbf{S}\} + \text{Im}\{\mathbf{N}_p\} \\ \text{Re}\{\mathbf{A}\Phi_x\mathbf{S}\} + \text{Re}\{\mathbf{N}_x\} \\ \text{Im}\{\mathbf{A}\Phi_x\mathbf{S}\} + \text{Im}\{\mathbf{N}_x\} \\ \text{Re}\{\mathbf{A}\Phi_y\mathbf{S}\} + \text{Re}\{\mathbf{N}_y\} \\ \text{Im}\{\mathbf{A}\Phi_y\mathbf{S}\} + \text{Im}\{\mathbf{N}_y\} \\ \text{Re}\{\mathbf{A}\Phi_z\mathbf{S}\} + \text{Re}\{\mathbf{N}_z\} \\ \text{Im}\{\mathbf{A}\Phi_z\mathbf{S}\} + \text{Im}\{\mathbf{N}_z\} \end{bmatrix} \quad (6.5.9)$$

矩阵 $V \in \mathbb{R}^{8M \times L}$ 可以表示为

$$V = \begin{bmatrix} \begin{bmatrix} \text{Re}\{\mathbf{A}\psi\} \\ \text{Im}\{\mathbf{A}\psi\} \end{bmatrix} \mathbf{S}_0 + \begin{bmatrix} \text{Re}\{\mathbf{N}_p\} \\ \text{Im}\{\mathbf{N}_p\} \end{bmatrix} \\ \begin{bmatrix} \text{Re}\{\mathbf{A}\psi\} \\ \text{Im}\{\mathbf{A}\psi\} \end{bmatrix} \Phi_x \mathbf{S}_0 + \begin{bmatrix} \text{Re}\{\mathbf{N}_x\} \\ \text{Im}\{\mathbf{N}_x\} \end{bmatrix} \\ \begin{bmatrix} \text{Re}\{\mathbf{A}\psi\} \\ \text{Im}\{\mathbf{A}\psi\} \end{bmatrix} \Phi_y \mathbf{S}_0 + \begin{bmatrix} \text{Re}\{\mathbf{N}_y\} \\ \text{Im}\{\mathbf{N}_y\} \end{bmatrix} \\ \begin{bmatrix} \text{Re}\{\mathbf{A}\psi\} \\ \text{Im}\{\mathbf{A}\psi\} \end{bmatrix} \Phi_z \mathbf{S}_0 + \begin{bmatrix} \text{Re}\{\mathbf{N}_z\} \\ \text{Im}\{\mathbf{N}_z\} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{A}}\mathbf{S}_0 \\ \bar{\mathbf{A}}\Phi_x\mathbf{S}_0 \\ \bar{\mathbf{A}}\Phi_y\mathbf{S}_0 \\ \bar{\mathbf{A}}\Phi_z\mathbf{S}_0 \end{bmatrix} + \tilde{\mathbf{N}} \quad (6.5.10)$$

其中

$$\bar{\mathbf{A}} = \begin{bmatrix} \text{Re}\{\mathbf{A}\psi\} \\ \text{Im}\{\mathbf{A}\psi\} \end{bmatrix} \in \mathbb{R}^{2M \times K} \quad (6.5.11)$$

定义矩阵 \mathbf{H}

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \cos\phi_1 \cos\varphi_1 & \cos\phi_2 \cos\varphi_2 & \cdots & \cos\phi_K \cos\varphi_K \\ \sin\phi_1 \cos\varphi_1 & \sin\phi_2 \cos\varphi_2 & \ddots & \sin\phi_K \cos\varphi_K \\ \sin\phi_1 & \sin\phi_2 & \cdots & \sin\phi_K \end{bmatrix} \in \mathbb{R}^{4 \times K} \quad (6.5.12)$$

式 (6.5.10) 中的矩阵 V_n ($n=1,2,3,4$) 可以表示为

$$V_n = \bar{\mathbf{A}}\mathbf{D}_n(\mathbf{H})\mathbf{S}_0 + \mathbf{N}_n, \quad n = 1, 2, 3, 4 \quad (6.5.13)$$

其中 $\mathbf{D}_n(\cdot)$ 是取矩阵的第 n 行作为对角线元素构造一个对角矩阵。 $\mathbf{D}_n(\mathbf{H}) = \text{diag}([h_{n,1}, h_{n,2}, \dots, h_{n,K}]) \in \mathbb{C}^{K \times K}$, $h_{m,k}$ 表示矩阵 \mathbf{H} 的第 (m, k) 个元素。式 (6.5.13) 中的无噪信号可以表示为三线性模型^[2,20]:

$$x_{m,l,n} = \sum_{k=1}^K a_{m,k} s_{l,k} h_{n,k}, \quad m = 1, \dots, 2M, \quad l = 1, \dots, L, \quad n = 1, 2, 3, 4 \quad (6.5.14)$$

其中 $s_{l,k}$ 是矩阵 \mathbf{S}_0 的第 (l, k) 个元素, 其余也如此。三线性模型展示了三个不同分集^[18]。式 (6.5.13) 可以解释为三维数据在空间某一方向的切片。根据式 (6.5.14) 的模型对称性, 矩阵重排后得到另外两个方向的切片:

$$\mathbf{U}_l = \mathbf{H} \mathbf{D}_p(\mathbf{S}_0^T) \bar{\mathbf{A}}^T + \mathbf{N}'_l, \quad l = 1, \dots, L \quad (6.5.15)$$

$$\mathbf{W}_m = \mathbf{S}_0^T \mathbf{D}_m(\bar{\mathbf{A}}) \mathbf{H}^T + \mathbf{N}'_m, \quad m = 1, \dots, 2M \quad (6.5.16)$$

其中 \mathbf{N}'_l 和 \mathbf{N}'_m 是接收噪声, \mathbf{V}_n 、 \mathbf{U}_l 和 \mathbf{W}_l 是三线性模型三个不同方向的切片。

注 6.5.1: PARAFAC 算法采用了 $M \times L \times 4$ 的三线性模型^[21], 而本节的算法中为 $2M \times L \times 4$ 的三线性模型。由于信号非圆性, 本节算法中的阵列孔径可以认为是 PARAFAC 算法中的两倍。

2. 三线性分解

TALS 是一种三线性模型的普遍数据检测方法^[20], 详细步骤如下。根据式 (6.5.10), 最小二乘 (LS) 准则为

$$\min_{\bar{\mathbf{A}}, \mathbf{H}, \mathbf{S}_0} \left\| \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \\ \mathbf{V}_4 \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{A}} \mathbf{D}_1(\mathbf{H}) \\ \bar{\mathbf{A}} \mathbf{D}_2(\mathbf{H}) \\ \bar{\mathbf{A}} \mathbf{D}_3(\mathbf{H}) \\ \bar{\mathbf{A}} \mathbf{D}_4(\mathbf{H}) \end{bmatrix} \mathbf{S}_0 \right\|_{\text{F}} \quad (6.5.17)$$

\mathbf{S}_0 的 LS 更新为

$$\hat{\mathbf{S}}_0 = \begin{bmatrix} \bar{\mathbf{A}} \mathbf{D}_1(\mathbf{H}) \\ \bar{\mathbf{A}} \mathbf{D}_2(\mathbf{H}) \\ \bar{\mathbf{A}} \mathbf{D}_3(\mathbf{H}) \\ \bar{\mathbf{A}} \mathbf{D}_4(\mathbf{H}) \end{bmatrix}^+ \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \\ \mathbf{V}_4 \end{bmatrix} \quad (6.5.18)$$

相似地, 第二个方向的切片 $\mathbf{U}_l = \mathbf{H} \mathbf{D}_l(\mathbf{S}_0^T) \bar{\mathbf{A}}^T + \mathbf{N}'_l$, $l = 1, \dots, L$, LS 准则为

$$\min_{\bar{\mathbf{A}}, \mathbf{H}, \mathbf{S}_0} \left\| \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_L \end{bmatrix} - \begin{bmatrix} \mathbf{H} \mathbf{D}_1(\mathbf{S}_0^T) \\ \mathbf{H} \mathbf{D}_2(\mathbf{S}_0^T) \\ \vdots \\ \mathbf{H} \mathbf{D}_L(\mathbf{S}_0^T) \end{bmatrix} \bar{\mathbf{A}}^T \right\|_{\text{F}} \quad (6.5.19)$$

$\bar{\mathbf{A}}$ 的 LS 更新为

$$\hat{\bar{\mathbf{A}}}^T = \begin{bmatrix} \mathbf{H} \mathbf{D}_1(\mathbf{S}_0^T) \\ \mathbf{H} \mathbf{D}_2(\mathbf{S}_0^T) \\ \vdots \\ \mathbf{H} \mathbf{D}_L(\mathbf{S}_0^T) \end{bmatrix}^+ \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_L \end{bmatrix} \quad (6.5.20)$$

最后, 第三个方向的切片 $\mathbf{W}_m = \mathbf{S}_0^T \mathbf{D}_m(\bar{\mathbf{A}}) \mathbf{H}^T + \mathbf{N}'_m$, $m = 1, \dots, 2M$, 则 \mathbf{H} 的 LS 更新为

$$\hat{\mathbf{H}}^T = \begin{bmatrix} \mathbf{S}_0^T \mathbf{D}_1(\bar{\mathbf{A}}) \\ \mathbf{S}_0^T \mathbf{D}_2(\bar{\mathbf{A}}) \\ \vdots \\ \mathbf{S}_0^T \mathbf{D}_{2M}(\bar{\mathbf{A}}) \end{bmatrix}^+ \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_{2M} \end{bmatrix} \quad (6.5.21)$$

三线性分解的残差平方和定义为

$$\text{SSR} = \sum_{m=1}^{2M} \sum_{l=1}^L \sum_{n=1}^4 \left| \tilde{x}_{m,l,n} - \sum_{k=1}^K \hat{a}_{m,k} \hat{s}_{l,k} \hat{h}_{n,k} \right|^2 \quad (6.5.22)$$

其中 $\tilde{x}_{m,l,n}$ 为含噪数据, $\hat{a}_{m,k}$ 、 $\hat{s}_{l,k}$ 和 $\hat{h}_{n,k}$ 分别是 $a_{m,k}$ 、 $s_{l,k}$ 和 $h_{n,k}$ 的估计值。

对于式 (6.5.18)、式 (6.5.20) 和式 (6.5.21), 矩阵 \mathbf{S}_0 、 $\bar{\mathbf{A}}$ 和 \mathbf{H} 分别采用最小二乘法更新。TALS 很容易实现并保证收敛性。在本节中, 采用复平行因子分析 (Complex Parallel Factor Analysis, COMFAC) 算法^[5]进行三线性分解, COMFAC 本质上就是 TALS 算法的一种快速实现方式。

3. 可辨识性

在这一部分, 讨论三线性分解的可辨识性。

定理 6.5.1^[2,17]: 考虑 $\mathbf{V}_n = \bar{\mathbf{A}}\mathbf{D}_n(\mathbf{H})\mathbf{S}_0 + \mathbf{N}_n$, 其中 $\bar{\mathbf{A}} \in \mathbb{R}^{2M \times K}$, $\mathbf{H} \in \mathbb{R}^{4 \times K}$, $\mathbf{S}_0 \in \mathbb{R}^{L \times K}$, 如果 $k_H + k_{S_0} + k_{\bar{A}} \geq 2K + 2$ (k_H 是 \mathbf{H} 的 k -秩), $\bar{\mathbf{A}}$, \mathbf{S}_0 和 \mathbf{H} 是唯一的 (存在尺度模糊、列模糊), 意味着另一组构成 \mathbf{V}_n 的 $\hat{\bar{\mathbf{A}}}, \hat{\mathbf{S}}_0, \hat{\mathbf{H}}$ ($n=1, \dots, 4$) 与 $\bar{\mathbf{A}}$ 、 \mathbf{S}_0 和 \mathbf{H} 的关系为

$$\hat{\bar{\mathbf{A}}} = \bar{\mathbf{A}}\boldsymbol{\Pi}\mathbf{A}_1 + \mathbf{N}_1, \quad \hat{\mathbf{S}}_0 = \mathbf{S}_0\boldsymbol{\Pi}\mathbf{A}_2 + \mathbf{N}_2, \quad \hat{\mathbf{H}} = \mathbf{H}\boldsymbol{\Pi}\mathbf{A}_3 + \mathbf{N}_3 \quad (6.5.23)$$

其中 $\boldsymbol{\Pi}$ 是一个列模糊矩阵, $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$ 是尺度模糊矩阵, 为对角矩阵且满足 $\mathbf{A}_1\mathbf{A}_2\mathbf{A}_3 = \mathbf{I}_K$ 。 \mathbf{N}_1 、 \mathbf{N}_2 和 \mathbf{N}_3 是估计误差。

一般情况下, 上面提到的矩阵 $\bar{\mathbf{A}}$ 、 \mathbf{S}_0 和 \mathbf{H} 都满足 k -秩, 因此可辨识条件变为

$$\min(4, K) + \min(L, K) + \min(2M, K) \geq 2K + 2 \quad (6.5.24)$$

4. 二维 DOA 估计

对于接收的含噪信号, 通过 PARAFAC 分解得到 $\hat{\mathbf{H}} = \mathbf{H}\boldsymbol{\Pi}\mathbf{A}_3 + \mathbf{N}_3$, 矩阵 $\hat{\mathbf{H}}$ 通过归一化处理消除了尺度模糊得到矩阵 $\hat{\mathbf{H}}$, $\hat{\mathbf{H}}$ 的第 i 列为 $\hat{\mathbf{h}}_i$ 。在无噪情况下, $\hat{\mathbf{h}}_i$ 为

$$\hat{\mathbf{h}}_i = \begin{bmatrix} 1 \\ \cos \phi_i \cos \varphi_i \\ \sin \phi_i \cos \varphi_i \\ \sin \varphi_i \end{bmatrix} \quad (6.5.25)$$

最终, 仰角和方位角的估计为

$$\hat{\phi}_i = \sin^{-1}(\hat{\mathbf{h}}_i(4)) \quad (6.5.26)$$

$$\hat{\phi}_i = \text{angle}(\hat{\mathbf{h}}_i(2) + j\hat{\mathbf{h}}_i(3)) \quad (6.5.27)$$

至此, 本节提出了声矢量传感器阵列下基于 NC-PARAFAC 的 2D-DOA 估计算法, 主要步骤如下。

步骤 1: 通过式 (6.5.8) 和式 (6.5.9) 构造矩阵 \mathbf{V} 。

步骤 2: 初始化矩阵 $\bar{\mathbf{A}}$ 、 \mathbf{S}_0 和 \mathbf{H} 。

步骤 3: 根据式 (6.5.18) 采用 LS 更新矩阵 \mathbf{S}_0 。

步骤 4: 根据式 (6.5.20) 采用 LS 更新矩阵 $\bar{\mathbf{A}}$ 。

步骤 5: 根据式 (6.5.21) 采用 LS 更新矩阵 \mathbf{H} 。

步骤 6: 重复步骤 3~5, 直到收敛。

步骤 7: 根据估计出 $\hat{\mathbf{H}}$ 来估计 2D-DOA。

注 6.5.2: 由于 $\hat{\phi}_i$ 和 $\hat{\phi}_i$ (仰角和方位角的估计值) 是通过 $\hat{\mathbf{h}}_i$ ($\hat{\mathbf{H}}$ 的第 i 列) 得到的, 因此估计出的 2D-DOA 可以实现自动配对。

注 6.5.3: 本节算法采用了随机初始化, 也可以利用某一方向的切片之间的旋转不变性和 ESPRIT 算法估计得到初始化值。

5. 复杂度分析

本节算法与 PARAFAC 算法相比具有更低的计算复杂度。PARAFAC 算法复杂度为 $O(nK^3 + 4nMLK)$, 其中 n 是 TALS 迭代次数, 而本节算法需要 $O(nK^3/4 + 2nMLK)$ 的复杂度。本节的算法的复杂度高于 ESPRIT 算法, ESPRIT 的复杂度为 $O(16M^2L + 64M^3 + K^3)$ 。图 6.5.2 是上述提到的三个算法的计算复杂度比较, 其中 $n=40$ 、 $K=3$ 和 $M=8$ 。从图 6.5.2 可以看出本节的算法复杂度是介于 ESPRIT 和 PARAFAC 算法之间的。

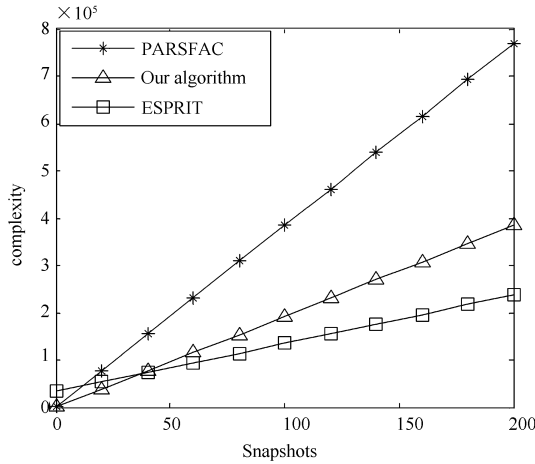


图 6.5.2 算法复杂度比较

本节所介绍的算法有以下优点:

- ① 算法的角度估计性能好于 ESPRIT、PM 和 PARAFAC 算法;
- ② 算法可以工作在传感器分布未知的任意声矢量阵列下;
- ③ 算法可以实现二维角度的自动配对;
- ④ 算法的计算复杂度低于 PARAFAC 算法。

6.5.3 CRB

非圆信号下的 DOA 估计的 CRB 与圆信号下有一些区别。需要估计的参数可以表示为

$$\boldsymbol{\zeta} = [\varphi_1, \dots, \varphi_K, \phi_1, \dots, \phi_K, \mathbf{s}_R^T(1), \dots, \mathbf{s}_R^T(L), \mathbf{s}_I^T(1), \dots, \mathbf{s}_I^T(L), \sigma^2]^T \quad (6.5.28)$$

其中 $\mathbf{s}_R(t), t=1, \dots, L$ 和 $\mathbf{s}_I(t), t=1, \dots, L$ 分别表示 $\mathbf{s}(t)$ 的实和虚部。

根据式 (6.5.6), L 个快拍的输出可以重写为

$$\mathbf{y} = [\mathbf{x}_E^T(1), \dots, \mathbf{x}_E^T(L), \mathbf{x}_E^H(1), \dots, \mathbf{x}_E^H(L)] \quad (6.5.29)$$

其中 $\mathbf{x}_E(l)$ 是矩阵 \mathbf{X}_E 的第 l 列。

\mathbf{y} 的均值 $\boldsymbol{\mu}$ 及协方差矩阵 $\boldsymbol{\Gamma}$ 分别为

$$\boldsymbol{\mu} = \begin{bmatrix} \mathbf{A}_E \mathbf{s}(1) \\ \vdots \\ \mathbf{A}_E \mathbf{s}(L) \\ \mathbf{A}_E^* \mathbf{s}^*(1) \\ \vdots \\ \mathbf{A}_E^* \mathbf{s}^*(L) \end{bmatrix}, \quad \boldsymbol{\Gamma} = \begin{bmatrix} \sigma^2 \mathbf{I} & & 0 \\ & \ddots & \\ 0 & & \sigma^2 \mathbf{I} \end{bmatrix} \quad (6.5.30)$$

根据文献[22], CRB 矩阵 \mathbf{P}_{cr} 的第 (i, j) 个元素可以表示为

$$[\mathbf{P}_{cr}^{-1}]_{ij} = \text{tr}[\boldsymbol{\Gamma}^{-1} \boldsymbol{\Gamma}'_i \boldsymbol{\Gamma}^{-1} \boldsymbol{\Gamma}'_j] + 2 \text{Re}[\boldsymbol{\mu}'_i^* \boldsymbol{\Gamma}^{-1} \boldsymbol{\mu}'_j] \quad (6.5.31)$$

其中 $\boldsymbol{\Gamma}'_i$ 和 $\boldsymbol{\mu}'_i$ 分别是 $\boldsymbol{\Gamma}$ 和 $\boldsymbol{\mu}$ 关于 $\boldsymbol{\zeta}$ 中第 i 个元素的导数。协方差矩阵与 σ^2 相关, 因此式 (6.5.31) 的第一部分可以被忽略。接着, CRB 矩阵 \mathbf{P}_{cr} 的第 (i, j) 个元素可以表示为

$$[\mathbf{P}_{cr}^{-1}]_{ij} = 2 \text{Re}[\boldsymbol{\mu}'_i^* \boldsymbol{\Gamma}^{-1} \boldsymbol{\mu}'_j] \quad (6.5.32)$$

根据式 (6.5.32)

$$\frac{\partial \boldsymbol{\mu}}{\partial \varphi_k} = \begin{bmatrix} \frac{\partial \mathbf{A}_E}{\partial \varphi_k} \mathbf{s}(1) \\ \vdots \\ \frac{\partial \mathbf{A}_E}{\partial \varphi_k} \mathbf{s}(L) \\ \frac{\partial \mathbf{A}_E^*}{\partial \varphi_k} \mathbf{s}^*(1) \\ \vdots \\ \frac{\partial \mathbf{A}_E^*}{\partial \varphi_k} \mathbf{s}^*(L) \end{bmatrix} = \begin{bmatrix} \mathbf{d}_{k\varphi} \mathbf{s}_k(1) \\ \vdots \\ \mathbf{d}_{k\varphi} \mathbf{s}_k(L) \\ \mathbf{d}_{k\varphi}^* \mathbf{s}_k^*(1) \\ \vdots \\ \mathbf{d}_{k\varphi}^* \mathbf{s}_k^*(L) \end{bmatrix}, \quad k=1, \dots, K \quad (6.5.33.a)$$

$$\frac{\partial \boldsymbol{\mu}}{\partial \phi_k} = \begin{bmatrix} \frac{\partial \mathbf{A}_E}{\partial \phi_k} \mathbf{s}(1) \\ \vdots \\ \frac{\partial \mathbf{A}_E}{\partial \phi_k} \mathbf{s}(L) \\ \frac{\partial \mathbf{A}_E^*}{\partial \phi_k} \mathbf{s}^*(1) \\ \vdots \\ \frac{\partial \mathbf{A}_E^*}{\partial \phi_k} \mathbf{s}^*(L) \end{bmatrix} = \begin{bmatrix} \mathbf{d}_{k\phi} \mathbf{s}_k(1) \\ \vdots \\ \mathbf{d}_{k\phi} \mathbf{s}_k(L) \\ \mathbf{d}_{k\phi}^* \mathbf{s}_k^*(1) \\ \vdots \\ \mathbf{d}_{k\phi}^* \mathbf{s}_k^*(L) \end{bmatrix}, \quad k=1, \dots, K \quad (6.5.33.b)$$

其中 $\mathbf{s}_k(t)$ 为 $\mathbf{s}(t)$ 的第 k 个元素，而

$$\mathbf{d}_{k\varphi} = \frac{\partial \mathbf{a}_E(\varphi_k, \phi_k)}{\partial \varphi_k} \quad (6.5.34.a)$$

$$\mathbf{d}_{k\phi} = \frac{\partial \mathbf{a}_E(\varphi_k, \phi_k)}{\partial \phi_k} \quad (6.5.34.b)$$

其中 $\mathbf{a}_E(\varphi_k, \phi_k)$ 是 \mathbf{A}_E 的第 k 列。

令

$$\mathbf{A} \triangleq \begin{bmatrix} \mathbf{d}_{1\varphi} \mathbf{s}_1(1) & \cdots & \mathbf{d}_{K\varphi} \mathbf{s}_K(1) & \mathbf{d}_{1\phi} \mathbf{s}_1(1) & \cdots & \mathbf{d}_{K\phi} \mathbf{s}_K(1) \\ \vdots & & \vdots & \vdots & & \vdots \\ \mathbf{d}_{1\varphi} \mathbf{s}_1(L) & \cdots & \mathbf{d}_{K\varphi} \mathbf{s}_K(L) & \mathbf{d}_{1\phi} \mathbf{s}_1(L) & \cdots & \mathbf{d}_{K\phi} \mathbf{s}_K(L) \end{bmatrix} \quad (6.5.35)$$

以及

$$\mathbf{G} \triangleq \begin{bmatrix} \mathbf{A}_E & 0 \\ & \ddots \\ 0 & \mathbf{A}_E \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} \mathbf{s}(1) \\ \vdots \\ \mathbf{s}(L) \end{bmatrix} \quad (6.5.36)$$

有 $\boldsymbol{\mu} = \begin{bmatrix} \mathbf{G}\mathbf{s} \\ \mathbf{G}^* \mathbf{s}^* \end{bmatrix}$ 和

$$\frac{\partial \boldsymbol{\mu}}{\partial \mathbf{s}_R^T} = \begin{bmatrix} \mathbf{G} \\ \mathbf{G}^* \end{bmatrix}, \quad \frac{\partial \boldsymbol{\mu}}{\partial \mathbf{s}_I^T} = \begin{bmatrix} j\mathbf{G} \\ -j\mathbf{G}^* \end{bmatrix} \quad (6.5.37)$$

根据式 (6.5.35) ~ 式 (6.5.37)

$$\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\zeta}^T} = \begin{bmatrix} \mathbf{A} & \mathbf{G} & j\mathbf{G} & 0 \\ \mathbf{A}^* & \mathbf{G}^* & -j\mathbf{G}^* & 0 \end{bmatrix} \quad (6.5.38)$$

式 (6.5.32) 可以表示为

$$2 \operatorname{Re} \left\{ \frac{\partial \boldsymbol{\mu}^*}{\partial \boldsymbol{\zeta}} \boldsymbol{\Gamma}^{-1} \frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\zeta}^T} \right\} = \begin{bmatrix} \mathbf{J} & 0 \\ 0 & 0 \end{bmatrix} \quad (6.5.39)$$

其中

$$\mathbf{J} \triangleq \frac{2}{\sigma^2} \operatorname{Re} \left\{ \begin{bmatrix} \mathbf{A}^H, \mathbf{A}^T \\ \mathbf{G}^H, \mathbf{G}^T \\ -j\mathbf{G}^H, j\mathbf{G}^T \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{G} & j\mathbf{G} \\ \mathbf{A}^* & \mathbf{G}^* & -j\mathbf{G}^* \end{bmatrix} \right\}$$

所以只需要考虑 \mathbf{J}^{-1} 中的元素。定义

$$\mathbf{B} \triangleq (\mathbf{G}^H \mathbf{G})^{-1} \mathbf{G}^H \mathbf{A} \quad (6.5.40)$$

$$\mathbf{F} \triangleq \begin{bmatrix} \mathbf{I} & 0 & 0 \\ -\mathbf{B}_R & \mathbf{I} & 0 \\ -\mathbf{B}_I & 0 & \mathbf{I} \end{bmatrix} \quad (6.5.41)$$

则

$$[\mathbf{A} \quad \mathbf{G} \quad j\mathbf{G}]\mathbf{F} = [(\mathbf{A} - \mathbf{G}\mathbf{B}) \quad \mathbf{G} \quad j\mathbf{G}] = [\Pi_G^\perp \mathbf{A} \quad \mathbf{G} \quad j\mathbf{G}] \quad (6.5.42)$$

$$[\mathbf{A}^* \quad \mathbf{G}^* \quad -j\mathbf{G}^*]\mathbf{F} = [(\mathbf{A}^* - \mathbf{G}^*\mathbf{B}^*) \quad \mathbf{G}^* \quad -j\mathbf{G}^*] = [\Pi_G^\perp \mathbf{A}^* \quad \mathbf{G}^* \quad -j\mathbf{G}^*] \quad (6.5.43)$$

其中 $\Pi_G^\perp = \mathbf{I} - \mathbf{G}(\mathbf{G}^H \mathbf{G})^{-1} \mathbf{G}^H$ 和 $\mathbf{G}^H \Pi_G^\perp = 0$ 。

$$\begin{aligned} \mathbf{F}^T \mathbf{J} \mathbf{F} &= \frac{2}{\sigma^2} \operatorname{Re} \left\{ \mathbf{F}^H \begin{bmatrix} \mathbf{A}^H, \mathbf{A}^T \\ \mathbf{G}^H, \mathbf{G}^T \\ -j\mathbf{G}^H, j\mathbf{G}^T \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{G} & j\mathbf{G} \\ \mathbf{A}^* & \mathbf{G}^* & -j\mathbf{G}^* \end{bmatrix} \mathbf{F} \right\} \\ &= \frac{2}{\sigma^2} \operatorname{Re} \left\{ \begin{bmatrix} \mathbf{A}^H \Pi_G^\perp, \mathbf{A}^T \Pi_G^\perp \\ \mathbf{G}^H, \mathbf{G}^T \\ -j\mathbf{G}^H, j\mathbf{G}^T \end{bmatrix} \begin{bmatrix} \Pi_G^\perp \mathbf{A} & \mathbf{G} & j\mathbf{G} \\ \Pi_G^\perp \mathbf{A}^* & \mathbf{G}^* & -j\mathbf{G}^* \end{bmatrix} \right\} \\ &= \frac{4}{\sigma^2} \operatorname{Re} \left\{ \begin{bmatrix} \mathbf{A}^H \Pi_G^\perp \mathbf{A} & 0 & 0 \\ 0 & \mathbf{G}^H \mathbf{G} & j\mathbf{G}^H \mathbf{G} \\ 0 & -j\mathbf{G}^H \mathbf{G} & \mathbf{G}^H \mathbf{G} \end{bmatrix} \right\} \end{aligned} \quad (6.5.44)$$

因此 \mathbf{J}^{-1} 可以写为

$$\begin{aligned} \mathbf{J}^{-1} &= \mathbf{F} (\mathbf{F}^T \mathbf{J} \mathbf{F})^{-1} \mathbf{F}^T \\ &= \frac{\sigma^2}{4} \begin{bmatrix} \mathbf{I} & 0 & 0 \\ -\mathbf{B}_R & \mathbf{I} & 0 \\ -\mathbf{B}_I & 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \operatorname{Re}(\mathbf{A}^H \Pi_G^\perp \mathbf{A}) & 0 & 0 \\ 0 & \kappa & \kappa \\ 0 & \kappa & \kappa \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{B}_R^T & -\mathbf{B}_I^T \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\sigma^2}{4} [\operatorname{Re}(\mathbf{A}^H \Pi_G^\perp \mathbf{A})]^{-1} & \kappa & \kappa \\ \kappa & \kappa & \kappa \\ \kappa & \kappa & \kappa \end{bmatrix} \end{aligned} \quad (6.5.45)$$

其中 κ 表示不感兴趣的部分。至此，给出 CRB 矩阵如下：

$$\operatorname{CRB} = \frac{\sigma^2}{4} [\operatorname{Re}(\mathbf{A}^H \Pi_G^\perp \mathbf{A})]^{-1} \quad (6.5.46)$$

通过进一步的简化，重写 CRB 矩阵：

$$\operatorname{CRB} = \frac{\sigma^2}{4L} \left\{ \operatorname{Re} \left[(\mathbf{D}^H \Pi_{A_E}^\perp \mathbf{D}) \oplus \mathbf{P}^T \right] \right\}^{-1} \quad (6.5.47)$$

其中 \oplus 表示 Hadamard 积， $\mathbf{P} = \frac{1}{L} \sum_{l=1}^L \mathbf{s}(l) \mathbf{s}^H(l)$ ， $\mathbf{D} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_K, \mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_K]$ ， $\mathbf{e}_k = \partial(\mathbf{a}(\varphi_k) \otimes \mathbf{h}_k) / \partial \varphi_k$ ， $\mathbf{f}_k = \partial(\mathbf{a}(\varphi_k) \otimes \mathbf{h}_k) / \partial \varphi_k$ ， $\mathbf{h}_k = [1, \mathbf{u}_k^T]^T$ ， $\Pi_{A_E}^\perp = \mathbf{I}_{4M} - \mathbf{A}_E (\mathbf{A}_E^H \mathbf{A}_E)^{-1} \mathbf{A}_E^H$ ， σ^2 为噪声功率。

6.5.4 仿真结果

定义均方根误差为

$$\text{RMSE} = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{N} \sum_{n=1}^N (\hat{a}_{k,n} - a_k)^2}$$

其中 $\hat{a}_{k,n}$ 是第 n 次 Monte Carlo 仿真中仰角/方位角 a_k 的估计值。假设 $K=3$ 个信源，其空间二维角度分别为 $(\phi_1, \varphi_1) = (10^\circ, 15^\circ)$ ， $(\phi_2, \varphi_2) = (30^\circ, 35^\circ)$ ， $(\phi_3, \varphi_3) = (50^\circ, 55^\circ)$ 。

图 6.5.3 和图 6.5.4 分别是 SNR=0dB 以及 12dB 下本节算法角度估计结果，其中 $M=10$ 、 $L=200$ 、 $M=10$ 和 $L=200$ 。从图 6.5.3 和图 6.5.4 中可以看出，本节算法可以有效估计出信源的 2D-DOA。

图 6.5.5 和图 6.5.6 是本节的 NC-PARAFAC 算法与 ESPRIT，PM 和 PARAFAC 算法的角度估计性能比较。从图 6.5.5 和图 6.5.6 中可以看出，本节算法的角度估计性能好于 ESPRIT 和 PM 算法，同时，由于利用了信号的非圆性质扩展阵列孔径，本节算法的角度估计性能好于 PARAFAC 算法，因此本节算法需要已知信号的非圆特性。

图 6.5.7 是在不同 L 值下本节算法的角度估计性能，其中 $M=10$ ， $K=3$ 。从图 6.5.7 中可以看出，本节算法的角度估计性能随着 L 的增加而变好。

图 6.5.8 是本节算法在不同 M 值下的角度估计性能，其中 $L=100$ ， $K=3$ 。从图 6.5.8 中可以看出，该算法性能随着阵列中声矢量传感器个数的增加而变优。

图 6.5.9 是本节算法对两个相近信源的角度估计。假设两个信源的空间二维角度为 $(\phi_1, \varphi_1) = (10^\circ, 15^\circ)$ ， $(\phi_2, \varphi_2) = (12^\circ, 17^\circ)$ 。在图 6.5.9 中， $M=12$ ， $J=100$ ， $K=2$ ，SNR=12dB。从图 6.5.9 中可以看出，本节算法可以有效估计相近信源的 DOA。

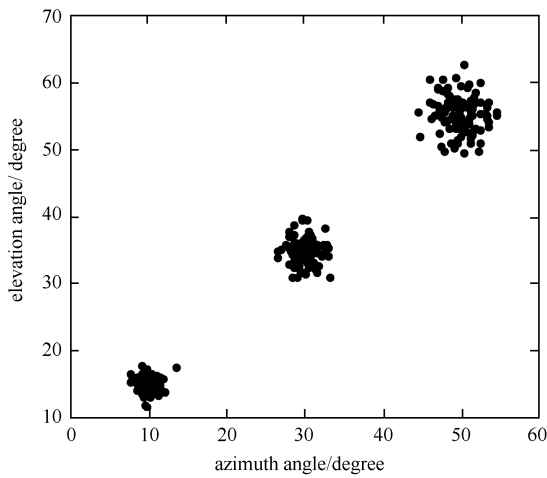


图 6.5.3 在 SNR=0dB 下本节算法的角度估计结果

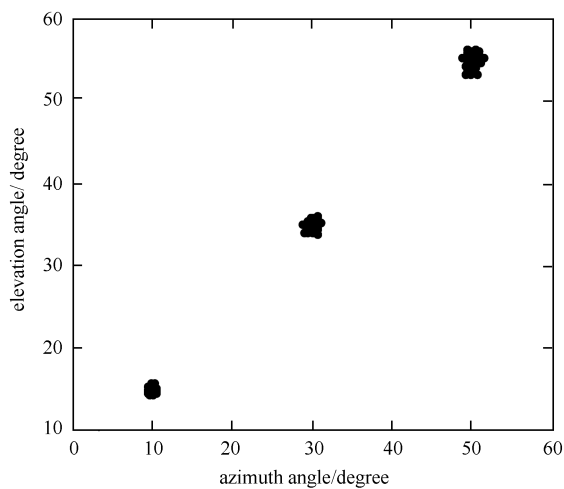
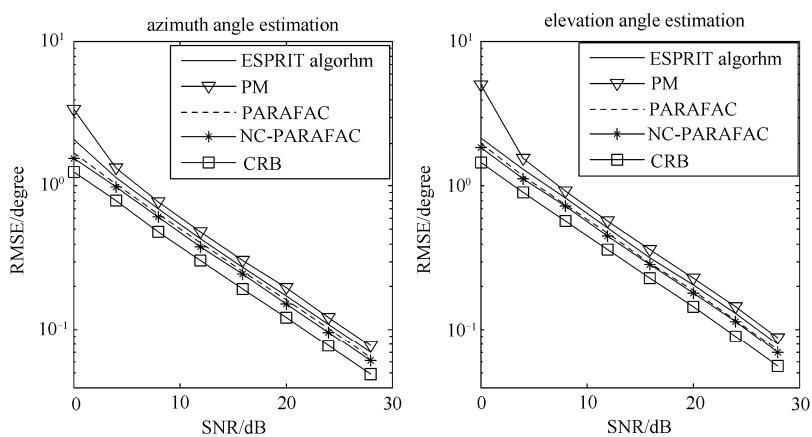
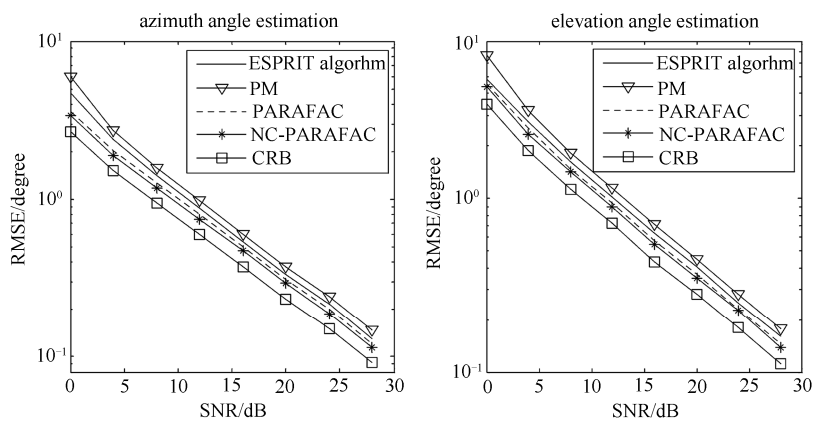


图 6.5.4 在 SNR=12dB 下本节算法的角度估计结果

图 6.5.5 算法的角度估计性能比较 ($M=8, L=200, K=3$)图 6.5.6 几个算法的角度估计性能比较 ($M=6, L=100, K=3$)

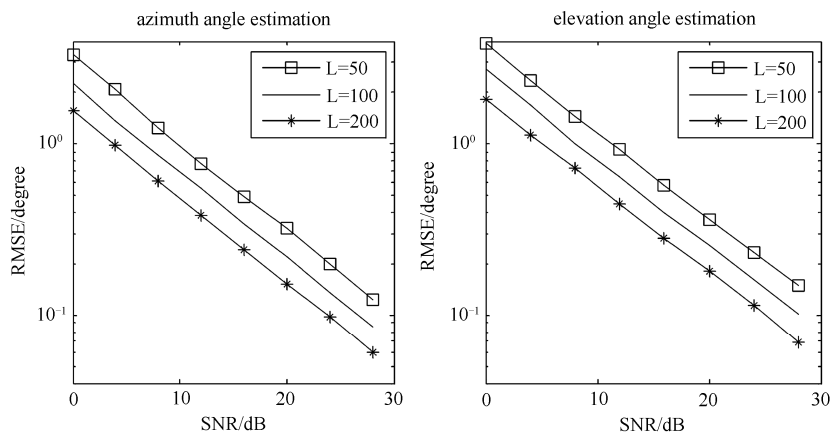


图 6.5.7 不同 L 值下本节算法的角度估计性能

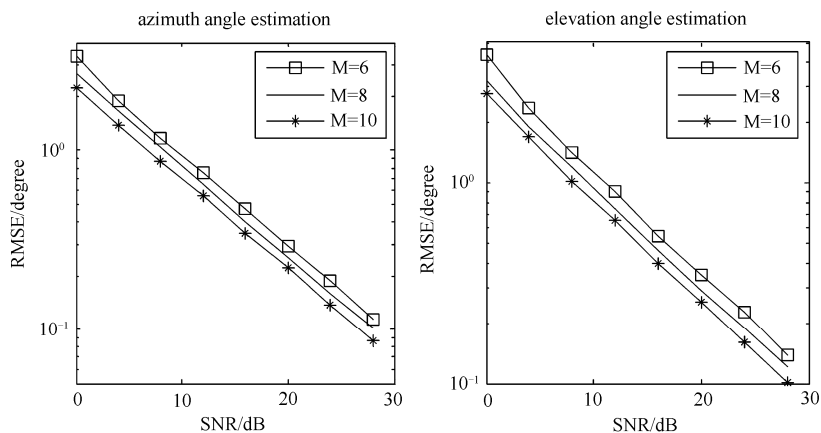


图 6.5.8 不同 M 值下本节算法的角度估计性能

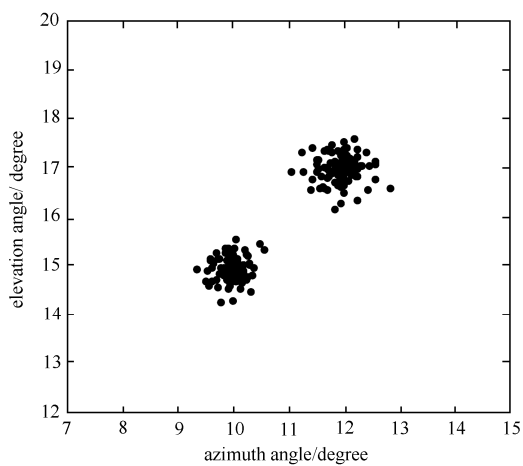


图 6.5.9 本节算法对相近信源的角度估计性能

参考文献

- [1] Xiaofei Zhang, Xin Gao, Dazhuan Xu. Novel Blind Carrier Frequency Offset Estimation for OFDM System with Multiple Antennas[J], IEEE Transactions on wireless communications, 2010, 9 (3) : 881-885.
- [2] J. B. Kruskal, Three-way arrays: Rank and uniqueness of trilinear decompositions with application to arithmetic complexity and statistics[C], Linear Algebra Applications, 1977, 18, pp. 95-138.
- [3] S.A. Vorobyov, Y. Rong, N.D. Sidiropoulos, and A.B. Gershman, Robust Iterative. Fitting of Multilinear Models[J], IEEE Transactions on Signal Processing, 2005, 53 (8) : 2678-2689.
- [4] G. Tomasi and R. Bro, A comparison of algorithms for fitting the PARAFAC model[C], Computational Statistics & Data Analysis, 2006, 50, pp.1700-1734.
- [5] R. Bro, N. D. Sidiropoulos, and G. B. Giannakis, A fast least squares algorithm for separating trilinear mixtures[C], in Proceedings of International Workshop Independent Component Analysis and Blind Signal Separation, Aussois, France, 1999, pp.289-294.
- [6] N. D. Sidiropoulos and X. Liu, Identifiability results for blind beamforming in incoherent multipath with small delay spread[J], IEEE Transactions on Signal Processing, 2001, 49 (1) :228-236.
- [7] Leshem, N. Petrochilos, and A. van der Veen, Finite sample identifiability of multiple constant modulus sources[C], in Proceeding of IEEE Workshop Sensor Array Multichannel Signal Process, 2002, pp. 408-412.
- [8] T. Jiang, and N.D. Sidiropoulos, Kruskal's permutation lemma and the identification of CANDECOMP/ PARAFAC and bilinear models with constant modulus[J], IEEE Transactions on Signal Processing, 2004, 52 (9) :2625-2636.
- [9] Zhang X.; Li J., Trilinear decomposition-based two-dimensional DOA estimation algorithm for arbitrarily spaced acoustic vector-sensor array subjected to unknown locations[J]. Wireless Personal Communication , 2012, 67: 859-877.
- [10] Xiaofei Zhang, Gaopeng Feng, Xin Gao, Dazhuan Xu, Blind multiuser detection for MC-CDMA with antenna array[J], Computers and Electrical Engineering,2010, 36 (1) : 160-168.
- [11] X. Zhang, F. Bao and D. Xu, Blind Joint Symbol Detection and DOA Estimation for OFDM System with Antenna Array[J]. Wireless Personal Communication, 2008, 46: 371-383.
- [12] D. Nion and N. D. Sidiropoulos. Adaptive algorithms to track the PARAFAC decomposition of a third-order tensor [J]. IEEE Transactions on Signal Processing, 2009,

57 (6) :2299-2310.

- [13] Hailang Wu, Xiaofei Zhang, Gaopeng Feng, et al. DOA Tracking in Monostatic MIMO Radar Using PARAFAC-RLST Algorithm [C]. The 3rd International Conference on Information Science and Engineering (ICISE2011) , 2011:958-961
- [14] 吴海浪. 单基地 MIMO 雷达的多目标角度估计和跟踪. 南京航空航天大学硕士学位论文, 2013.
- [15] 张怀根, 张林让, 吴顺君等. 一种新的多目标角跟踪方法[J]. 电子与信息学报, 2007, 29 (12) :2840-2842.
- [16] C. R. Rao, L. Zhang, and L. C. Zhao. Multitarget angle tracking algorithm for data association [J]. IEEE Transactions on Signal Processing, 1994, 42 (2) :459 -462.
- [17] Zhang X, Cao R, Zhou M. Noncircular-PARAFAC for 2D-DOA estimation of noncircular signals in arbitrarily spaced acoustic vector-sensor array subjected to unknown locations[J]. EURASIP Journal on Advances in Signal Processing, 2013, 2013 (1) : 1-10.
- [18] Wong K T, Zoltowski M D. Closed-form underwater acoustic direction-finding with arbitrarily spaced vector hydrophones at unknown locations[J]. IEEE Journal of Oceanic Engineering, 1997, 22 (3) : 566-575.
- [19] Wong K T, Zoltowski M D. Extended-aperture underwater acoustic multisource azimuth/elevation direction-finding using uniformly but sparsely spaced vector hydrophones[J]. IEEE Journal of Oceanic Engineering, 1997, 22 (4) : 659-672.
- [20] Sidiropoulos N D, Bro R, Giannakis G B. Parallel factor analysis in sensor array processing[J]. IEEE Transactions on Signal Processing, 2000, 48 (8) : 2377-2388.
- [21] Zhang X, Li J, Chen H, et al. Trilinear decomposition-based two-dimensional DOA estimation algorithm for arbitrarily spaced acoustic vector-sensor array subjected to unknown locations[J]. Wireless Personal Communications, 2012, 67 (4) : 859-877.
- [22] Stoica P, Nehorai A. Performance study of conditional and unconditional direction-of-arrival estimation[J]. IEEE Transactions on Acoustics, Speech and Signal Processing, 1990, 38 (10) : 1783-1795.

第7章

四线性分解在通信和信号处理中的应用

本章就四线性分解理论在通信和信号处理中的应用，给出几个例子。

7.1 基于四线性分解的均匀面阵的角度和频率联合估计

本节针对面阵提出基于平行因子四线性分解^[1, 2]的信号二维角度和频率估计算法^[3]，将平行因子四线性分解法应用到均匀面阵进行研究，并与现有的算法进行比较和分析，最后用仿真验证算法的有效性。

7.1.1 数据模型

设平面阵列阵元数为 $M \times N$ ，其中， M 为面阵的行数， N 为列数。平面阵列的阵元分布为二维分布，每个阵元分布在 X - Y 平面上，假设每行为 X 轴取向，行上阵元以等距离 Δx 分布，每列则为 Y 轴取向，列上阵元以等距离 Δy 分布。设 $\Delta x = \Delta y = d$ ，假定有 K 个彼此独立且到达角均不同的窄带信号入射到该阵列上，考虑入射信号位于远源场，第 k 个信号的二维到达角为 (θ_k, ϕ_k) ， $k=1, 2, \dots, K$ ，其中 θ_k 、 ϕ_k 分别代表第 k 个信源的仰角和方位角，如图 7.1.1 所示。以原点为参考点，则阵列

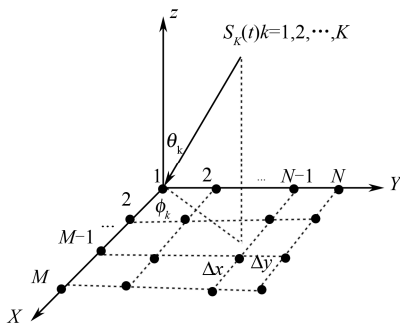


图 7.1.1 平面阵列

中第 (i, j) 个阵元的输出信号为

$$\mathbf{x}_{ij}(t) = \sum_{k=1}^K \mathbf{a}(\theta_k, \phi_k, f_k) s_k(t) + w_{ij}(t) \quad (7.1.1)$$

式中 f_k 为信号未知频率, x_i 、 y_j 分别为阵元在 X - Y 平面分布的横坐标和纵坐标。

$$\mathbf{a}(\theta_k, \phi_k, f_k) = \exp(-j2\pi(x_i \cos \phi_k \sin \theta_k + y_j \sin \phi_k \sin \theta_k) f_k / c) \quad (7.1.2)$$

为了对频率进行估计, 对第 (i, j) 个阵元接收信号加上时延 τ_p ($p=1, 2, \dots, P$) 得到

$$\begin{aligned} \mathbf{x}_{ij}(t - \tau_p) &= \sum_{k=1}^K \mathbf{a}(\theta_k, \phi_k, f_k) s_k(t - \tau_p) + w_{ij}(t - \tau_p) \\ &= \sum_{k=1}^K \mathbf{a}(\theta_k, \phi_k, f_k) s_k(t) e^{-j2\pi f_k \tau_p} + w_{ij}(t - \tau_p) \end{aligned} \quad (7.1.3)$$

假设空时信道在 L 个符号内恒定不变, 上述阵列天线输出可表示为

$$\mathbf{X} = \mathbf{A} \mathbf{S}^T + \mathbf{W} \quad (7.1.4)$$

其中

$$\mathbf{A} = [\mathbf{a}(\theta_1, \phi_1, f_1) \quad \mathbf{a}(\theta_2, \phi_2, f_2) \quad \dots \quad \mathbf{a}(\theta_K, \phi_K, f_K)]^T \in \mathbb{C}^{MN \times K}$$

$$\mathbf{S} = [\mathbf{s}_1 \quad \mathbf{s}_2 \quad \dots \quad \mathbf{s}_K]^T \in \mathbb{C}^{L \times K}$$

公式 (7.1.4) 中的信号经过时延 τ_p 后可以表示为

$$\mathbf{X}_p = \mathbf{A} \Phi_p \mathbf{S}^T, \quad p=1, 2, \dots, P+1 \quad (7.1.5)$$

其中

$$\Phi_p = \begin{bmatrix} e^{-j2\pi f_1 \tau_p} & & & \\ & e^{-j2\pi f_2 \tau_p} & & \\ & & \ddots & \\ & & & e^{-j2\pi f_K \tau_p} \end{bmatrix} \in \mathbb{C}^{K \times K}$$

定义时延矩阵

$$\Phi = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{-j2\pi f_1 \tau_1} & e^{-j2\pi f_2 \tau_1} & \dots & e^{-j2\pi f_K \tau_1} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j2\pi f_1 \tau_P} & e^{-j2\pi f_2 \tau_P} & \dots & e^{-j2\pi f_K \tau_P} \end{bmatrix} \in \mathbb{C}^{(P+1) \times K} \quad (7.1.6)$$

公式 (7.1.5) 也可以表示为

$$\mathbf{X}_p = \mathbf{A} \mathbf{D}_p(\Phi) \mathbf{S}^T + \mathbf{W}_p, \quad p=1, 2, \dots, P+1 \quad (7.1.7)$$

7.1.2 平行因子四线性模型形成

假设阵元间距均相等, 令

$$\mathbf{a}_1(\theta_k, \phi_k, f_k) = [1, e^{-j2\pi df_k \cos \phi_k \sin \theta_k / c}, \dots, e^{-j2\pi d(M-1)f_k \cos \phi_k \sin \theta_k / c}]^T \quad (7.1.8)$$

$$\mathbf{a}_2(\theta_k, \phi_k, f_k) = [1, e^{-j2\pi df_k \sin \phi_k \sin \theta_k / c}, \dots, e^{-j2\pi df_k(N-1) \sin \phi_k \sin \theta_k / c}]^T \quad (7.1.9)$$

因为每一列可看成一个均匀线阵, 令

$$\mathbf{A}_1 = [\mathbf{a}_1(\theta_1, \phi_1, f_1), \mathbf{a}_1(\theta_2, \phi_2, f_2), \dots, \mathbf{a}_1(\theta_K, \phi_K, f_K)] \in \mathbb{C}^{M \times K} \quad (7.1.10)$$

则第一列上的接收信号可以表示为

$$\mathbf{X}_1 = \mathbf{A}_1 \mathbf{D}_p(\boldsymbol{\Phi}) \mathbf{S}^T + \mathbf{W}_1, \quad p=1, 2, \dots, P+1 \quad (7.1.11)$$

第二列上的接收信号表示为

$$\mathbf{X}_2 = \mathbf{A}_1 \text{diag} \begin{bmatrix} e^{-j2\pi df_1 \sin \phi_1 \sin \theta_1 / c} \\ e^{-j2\pi df_2 \sin \phi_2 \sin \theta_2 / c} \\ \vdots \\ e^{-j2\pi df_K \sin \phi_K \sin \theta_K / c} \end{bmatrix} \mathbf{D}_p(\boldsymbol{\Phi}) \mathbf{S}^T + \mathbf{W}_2, \quad p=1, 2, \dots, P+1 \quad (7.1.12)$$

以此类推, 第 N 列上的接收信号表示为

$$\mathbf{X}_N = \mathbf{A}_1 \text{diag} \begin{bmatrix} e^{-j2\pi df_1(N-1) \sin \phi_1 \sin \theta_1 / c} \\ e^{-j2\pi df_2(N-1) \sin \phi_2 \sin \theta_2 / c} \\ \vdots \\ e^{-j2\pi df_K(N-1) \sin \phi_K \sin \theta_K / c} \end{bmatrix} \mathbf{D}_p(\boldsymbol{\Phi}) \mathbf{S}^T + \mathbf{W}_N, \quad p=1, 2, \dots, P+1 \quad (7.1.13)$$

定义

$$\mathbf{A}_2 = [\mathbf{a}_2(\theta_1, \phi_1, f_1), \mathbf{a}_2(\theta_2, \phi_2, f_2), \dots, \mathbf{a}_2(\theta_K, \phi_K, f_K)] \in \mathbb{C}^{N \times K} \quad (7.1.14)$$

由上述推导可知阵列接收信号可以表示为

$$\mathbf{X} = [\mathbf{A}_2 \odot \mathbf{A}_1] \mathbf{D}_p(\boldsymbol{\Phi}) \mathbf{S}^T + \mathbf{W}_p, \quad p=1, 2, \dots, P+1 \quad (7.1.15)$$

其中 \odot 为 Khatri-Rao 积, 上式可进一步表示为

$$\mathbf{X}_{np} = \mathbf{A}_1 \mathbf{D}_n(\mathbf{A}_2) \mathbf{D}_p(\boldsymbol{\Phi}) \mathbf{S}^T + \mathbf{W}_{np}, \quad n=1, 2, \dots, N, \quad p=1, 2, \dots, P+1 \quad (7.1.16)$$

将 $N \times (P+1)$ 个切片积累成 $M \times (P+1) \times N \times L$ 的四维数据集 \mathbf{X} , 则式 (7.1.15) 有以下形式:

$$x_{n,m,p,l} = \sum_{k=1}^K s_{l,k} \psi_{p,k} \alpha_{m,k} b_{n,k} \quad (7.1.17)$$

$$n=1, 2, \dots, N, \quad m=1, 2, \dots, M, \quad p=1, \dots, P+1, \quad l=1, 2, \dots, L$$

其中 $a_{m,k}$ 是 \mathbf{A}_1 矩阵中第 (m,k) 个元素, $b_{n,k}$ 是 \mathbf{A}_2 矩阵中第 (n,k) 个元素, $\psi_{p,k}$ 是矩阵 $\boldsymbol{\Phi}$ 中第 (p,k) 个元素, $s_{l,k}$ 是 \mathbf{S} 中第 (l,k) 个元素。上式称为四线性模型, 根据其对称性, 可写成其他三种形式:

$$Y_{mn} = S D_m(A_1) D_n(A_2) \Phi^T, \quad m=1,2,\dots,M, \quad n=1,2,\dots,N \quad (7.1.18)$$

$$Z_{lm} = \Phi D_l(S) D_m(A_1) A_2^T, \quad l=1,2,\dots,L, \quad m=1,2,\dots,M \quad (7.1.19)$$

$$W_{pl} = A_2 D_p(\Phi) D_l(S) A_1^T, \quad p=1,2,\dots,P+1, \quad l=1,2,\dots,L \quad (7.1.20)$$

下面讨论其可辨识性。

定理 7.1.1: $X_{np} = A_1 D_n(A_2) D_p(\Phi) S^T + W_{np}$, ($n=1,2,\dots,M$, $p=1,2,\dots,P+1$), 其中 $A_1 \in \mathbb{C}^{M \times K}$, $A_2 \in \mathbb{C}^{N \times K}$, $\Phi \in \mathbb{C}^{(P+1) \times K}$, $S \in \mathbb{C}^{K \times L}$, 如果矩阵 A_1 、 A_2 、 Φ 、 S 的 k -满足下面的条件:

$$k_{A_1} + k_{A_2} + k_\Phi + k_S \geq 2K + 3 \quad (7.1.21)$$

则 A_1 、 A_2 、 Φ 和 S 对于列交换和 (复数) 尺度变换是唯一的, 即 A_1 、 A_2 、 Φ 、 S 可唯一确定。 X 的 k -秩分解具有唯一性。

7.1.3 算法描述

1. 平行因子模型分解

本节采用 QALS 算法对式(7.1.16)给出的平行因子模型进行拟合。式(7.1.18)、式(7.1.19)、式(7.1.20)给出了平行因子模型其他形式, 它们有着相同的元素, 只是排列顺序不同而已。迭代过程中的代价函数为

$$\min_{A_1, A_2, S, \Phi} \left\| \begin{bmatrix} \tilde{X}_{11} \\ \tilde{X}_{12} \\ \vdots \\ \tilde{X}_{N(P+1)} \end{bmatrix} - \begin{bmatrix} A_1 D_1(A_2) D_1(\Phi) \\ A_1 D_1(A_2) D_2(\Phi) \\ \vdots \\ A_1 D_N(A_2) D_{P+1}(\Phi) \end{bmatrix} S^T \right\|_F \quad (7.1.22)$$

给定 X 、 A_1 、 A_2 、 Φ 、 S 的最小二乘解为

$$\hat{S}^T = \begin{bmatrix} \hat{A}_1 D_1(\hat{A}_2) D_1(\hat{\Phi}) \\ \hat{A}_1 D_1(\hat{A}_2) D_2(\hat{\Phi}) \\ \vdots \\ \hat{A}_1 D_N(\hat{A}_2) D_{P+1}(\hat{\Phi}) \end{bmatrix}^\dagger \begin{bmatrix} \tilde{X}_{11} \\ \tilde{X}_{12} \\ \vdots \\ \tilde{X}_{N(P+1)} \end{bmatrix} \quad (7.1.23)$$

同样利用式(7.1.18), 给定 Y 、 S 、 A_1 、 A_2 、 Φ 的最小二乘解为

$$\hat{\Phi}^T = \begin{bmatrix} \hat{S} D_1(\hat{A}_1) D_1(\hat{A}_2) \\ \hat{S} D_1(\hat{A}_1) D_2(\hat{A}_2) \\ \vdots \\ \hat{S} D_M(\hat{A}_1) D_N(\hat{A}_2) \end{bmatrix}^\dagger \begin{bmatrix} \tilde{Y}_{11} \\ \tilde{Y}_{12} \\ \vdots \\ \tilde{Y}_{MN} \end{bmatrix} \quad (7.1.24)$$

同样利用式(7.1.19), 给定 Z 、 Φ 、 S 、 A_1 、 A_2 的最小二乘解为

$$\hat{\mathbf{A}}_2^T = \begin{bmatrix} \hat{\Phi} \mathbf{D}_1(\hat{\mathbf{S}}) \mathbf{D}_1(\hat{\mathbf{A}}_1) \\ \hat{\Phi} \mathbf{D}_1(\hat{\mathbf{S}}) \mathbf{D}_2(\hat{\mathbf{A}}_1) \\ \vdots \\ \hat{\Phi} \mathbf{D}_L(\hat{\mathbf{S}}) \mathbf{D}_M(\hat{\mathbf{A}}_1) \end{bmatrix}^T \begin{bmatrix} \tilde{\mathbf{Z}}_{11} \\ \tilde{\mathbf{Z}}_{12} \\ \vdots \\ \tilde{\mathbf{Z}}_{LM} \end{bmatrix} \quad (7.1.25)$$

同样利用式 (7.1.20)，给定 \mathbf{W} 、 Φ 、 \mathbf{S} 、 \mathbf{A}_2 、 \mathbf{A}_1 的最小二乘解为

$$\hat{\mathbf{A}}_1^T = \begin{bmatrix} \hat{\mathbf{A}}_2 \mathbf{D}_1(\hat{\Phi}) \mathbf{D}_1(\hat{\mathbf{S}}) \\ \hat{\mathbf{A}}_2 \mathbf{D}_1(\hat{\Phi}) \mathbf{D}_2(\hat{\mathbf{S}}) \\ \vdots \\ \hat{\mathbf{A}}_2 \mathbf{D}_{P+1}(\hat{\Phi}) \mathbf{D}_L(\hat{\mathbf{S}}) \end{bmatrix}^T \begin{bmatrix} \tilde{\mathbf{W}}_{11} \\ \tilde{\mathbf{W}}_{12} \\ \vdots \\ \tilde{\mathbf{W}}_{(P+1)L} \end{bmatrix} \quad (7.1.26)$$

利用上述方法，每次更新一个矩阵，本次更新的矩阵立刻参与下一个矩阵的求解，下面给出算法的具体实现。

步骤 1：初始化方向矩阵 \mathbf{A}_1 和 \mathbf{A}_2 、时延矩阵 Φ 、信源矩阵 \mathbf{S} ， $k=0$ 。

步骤 2： $k=k+1$ 。

步骤 3：利用式 (7.1.23) 计算 \mathbf{S} 的第 k 次迭代值。

步骤 4：利用式 (7.1.24) 计算 Φ 的第 k 次迭代值。

步骤 5：利用式 (7.1.25) 计算 \mathbf{A}_2 的第 k 次迭代值。

步骤 6：利用式 (7.1.26) 计算 \mathbf{A}_1 的第 k 次迭代值。

步骤 7：重复步骤 2~6，直至算法收敛。

估计出的 $\hat{\mathbf{A}}_1$ 、 $\hat{\mathbf{A}}_2$ 、 $\hat{\Phi}$ 含有信源的角度和频率信息。根据 PARAFAC 模型的唯一性定理可知， Φ 、 \mathbf{A}_1 和 \mathbf{A}_2 矩阵的估计值均有相同的列交换矩阵，这意味着 $\hat{\Phi}$ 第 f 列估计的频率和方向矩阵 $\hat{\mathbf{A}}_1$ 、 $\hat{\mathbf{A}}_2$ 的第 f 列相对应，自然实现了角度和频率的配对，这是传统的算法所不具备的。

上述算法结束后，得到时延矩阵 Φ 的估计值 $\hat{\Phi}$ 可表示成如下的形式：

$$\hat{\Phi} = \Phi \Pi \Delta + \mathbf{N}$$

Π 是列模糊矩阵， Δ 为尺度模糊矩阵， \mathbf{N} 指所估计的误差矩阵。矩阵 $\hat{\Phi}$ 的第 i 列为 $\hat{\mathbf{g}}(f_i)$ 。在没有噪声的情况下， $\hat{\mathbf{g}}(f_i)$ 可表示为

$$\hat{\mathbf{g}}(f_i) = [1, e^{-j2\pi f_i \tau_1}, \dots, e^{-j2\pi f_i \tau_p}]^T \quad (7.1.27)$$

可以得到

$$\hat{\mathbf{h}} = -\text{angle}(\mathbf{g}(f_i)) = [0, 2\pi f_i \tau_1, \dots, 2\pi f_i \tau_p]^T \quad (7.1.28)$$

令

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 0 \\ 1 & 2\pi\tau_1 \\ \vdots & \vdots \\ 1 & 2\pi\tau_P \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b_0 \\ f_i \end{bmatrix}$$

则

$$\mathbf{P}_1 \mathbf{B} = \hat{\mathbf{h}} \quad (7.1.29)$$

f_i 的最小二乘结果为

$$\hat{f}_i = (\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T \hat{\mathbf{h}} \quad (7.1.30)$$

根据式 (7.1.28) ~ 式 (7.1.30), 可依次得到 K 个用户的频率估计。

2. 二维方向角的估计

同样, 可得到方向估计矩阵 $\hat{\mathbf{A}}_1$, $\hat{\mathbf{A}}_2$, 它们的第 k 列为

$$\hat{\mathbf{a}}_1(\theta_k, \phi_k, f_k) = [1, e^{-j2\pi d \cos \phi_k \sin \theta_k f_k / c}, \dots, e^{-j2\pi d (M-1) \cos \phi_k \sin \theta_k f_k / c}]^T \quad (7.1.31)$$

$$\hat{\mathbf{a}}_2(\theta_k, \phi_k, f_k) = [1, e^{-j2\pi d \sin \phi_k \sin \theta_k f_k / c}, \dots, e^{-j2\pi d (N-1) \sin \phi_k \sin \theta_k f_k / c}]^T \quad (7.1.32)$$

对上两式取对数的虚部得到:

$$\begin{aligned} \hat{\mathbf{g}}_1 &= -\text{angle}((\mathbf{a}_1(\theta_k, \phi_k, f_k))) \\ &= [0, 2\pi d \cos \phi_k \sin \theta_k f_k / c, \dots, 2\pi d (M-1) \cos \phi_k \sin \theta_k f_k / c]^T \\ &= [0, mf_k / c, 2mf_k / c, \dots, (M-1)mf_k / c]^T \end{aligned} \quad (7.1.33)$$

$$\begin{aligned} \hat{\mathbf{g}}_2 &= -\text{angle}(\mathbf{a}_2(\theta_k, \phi_k)) \\ &= [0, 2\pi d \sin \phi_k \sin \theta_k f_k / c, \dots, 2\pi d (N-1) \sin \phi_k \sin \theta_k f_k / c]^T \\ &= [0, nf_k / c, 2nf_k / c, \dots, (N-1)nf_k / c]^T \end{aligned} \quad (7.1.34)$$

令

$$\mathbf{P}_2 = \begin{bmatrix} 1 & 0 \\ 1 & \hat{f}_k / c \\ \vdots & \vdots \\ 1 & (M-1)\hat{f}_k / c \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} m_0 \\ m \end{bmatrix}$$

则

$$\mathbf{P}_2 \mathbf{C} = \hat{\mathbf{g}}_1 \quad (7.1.35)$$

\hat{f}_k 是通过式 (7.1.30) 得到的频率估计, m 是 $2\pi d \sin \theta_k \cos \phi_k$ 的估计值, m 的最小二乘结果为

$$\begin{bmatrix} \hat{m}_0 \\ \hat{m} \end{bmatrix} = (\mathbf{P}_2^T \mathbf{P}_2)^{-1} \mathbf{P}_2^T \hat{\mathbf{g}}_1 \quad (7.1.36)$$

同理可得到 \hat{n} , 即 $2\pi d \sin \theta_k \sin \phi_k$ 的估计值。所以DOA估计为

$$\hat{\theta}_k = \sin^{-1} \left(\sqrt{\hat{m}^2 + \hat{n}^2} / (2\pi d) \right) \quad (7.1.37)$$

$$\hat{\phi}_k = \tan^{-1}(\hat{n} / \hat{m}) \quad (7.1.38)$$

7.1.4 仿真结果

仿真 1: 为了验证本节提出的算法的性能, 采用 Monte Carlo 仿真来评估算法的角度和频率估计性能, 仿真中采用阵列为 4×4 的均匀面阵。假设非相关窄带信号源数为 $K=3$, 其二维 DOA 和频率分别为 $(15^\circ, 10^\circ, 1.2\text{MHz})$, $(25^\circ, 20^\circ, 1.5\text{MHz})$, $(35^\circ, 30^\circ, 1.8\text{MHz})$, 阵元间距 $\Delta x = \Delta y = 75\text{m}$, 快拍数 $L=500$ 。图 7.1.2 给出了 $\text{SNR}=10\text{dB}$ 时方位角和仰角的散布图及方位角和频率的散布图, 从图 7.1.2 可以看出此算法具有较好的联合角度和频率估计性能。

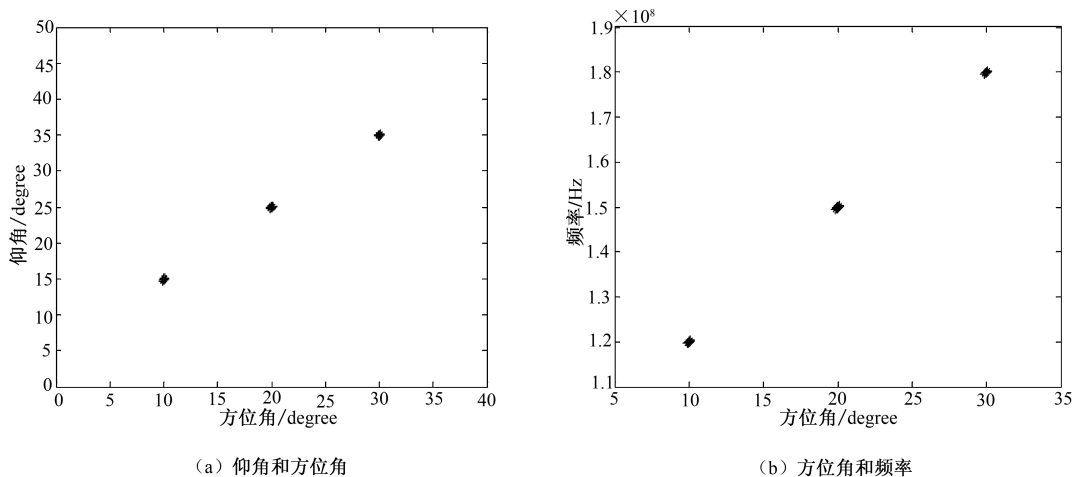


图 7.1.2 $\text{SNR}=10\text{dB}$ 的二维角度和频率散布图

仿真 2: 验证本节算法在不同信噪比下的角度和频率参数联合估计性能, 仿真参数同上。定义 RMSE:

$$\text{RMSE} = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{500} \sum_{m=1}^{500} (a_{mk} - a_{ok})^2} \quad (7.1.39)$$

其中, a_{mk} 为第 k 个信源的第 m 个 Monte Carlo 仿真的角度或频率的估计值, a_{ok} 为第 k 个信源的角度或频率的真实值。为了进行比较, ESPRIT 算法和基于三线性 (trilinear) 分解算法的结果同时在图中给出。图 7.1.3~图 7.1.5 分别为在不同信噪比条件下的频率和二维 DOA 估计值的 RMSE 曲线, 可以看出, 随着信噪比的提高, RMSE 变小, 估计性能更好。同时可看出本节算法的性能优于其他两种算法, trilinear 分解算法角度估计性能又优于 ESPRIT 算法。

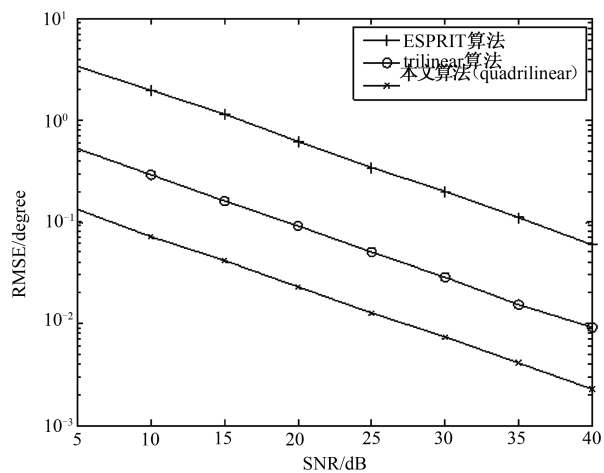


图 7.1.3 仰角估计性能对比图

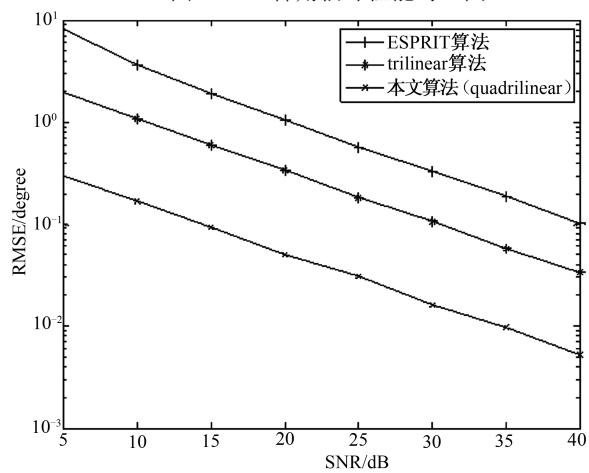


图 7.1.4 方位角估计性能对比图

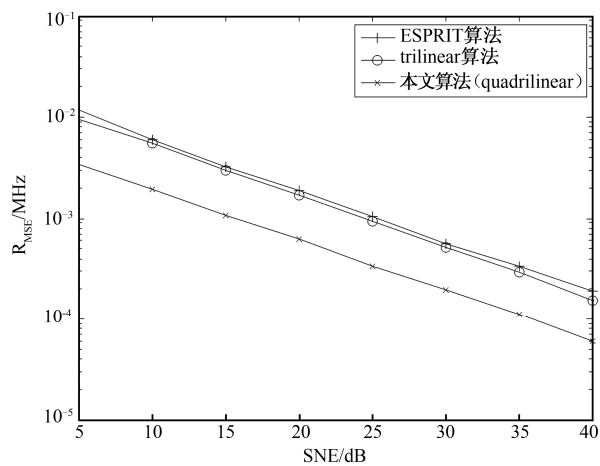


图 7.1.5 频率估计性能对比图

7.2 基于四线性分解的双基地 MIMO 雷达的角度和多普勒频率联合估计

本节研究双基地多输入多输出 (Multiple-Input Multiple-Output, MIMO) 雷达中的角度和多普勒频率联合估计问题, 并提出一种基于四线性分解 (Quadrilinear Decomposition) 的离开角 (Direction Of Departure, DOD)、波达角 (Direction Of Arrival, DOA) 和多普勒频率的联合估计算法^[4]。通过对接收端匹配滤波器的输出进行延迟操作, 得到符合四线性模型的数据, 根据四线性交替最小二乘 (Quadrilinear Alternating Least Squares, QALS) 进行迭代, 得到方向矩阵和多普勒频率矩阵的估计, 进而得到角度和频率的估计。该算法无须谱峰搜索, 无须知道反射系数, 可实现角度和频率的自动配对, 且能用于非均匀阵。而且该算法的角度估计性能优于多维 ESPRIT 方法和三线性分解方法。同时, 分析本节算法的复杂度, 并推导出克拉美-罗界 (Cramer-Rao Bound, CRB)。最后用仿真结果验证该算法的有效性。

7.2.1 双基地 MIMO 雷达时空数据模型

考虑双基地 MIMO 雷达中, 发射阵列为 M 元的线阵, 分布于 y 轴, 以第 1 个阵元为参考阵元, 第 m 个阵元和其之间的距离设为 $d_m^M (m=2, \dots, M)$; 接收阵列为 N 元的线阵, 也分布于 y 轴, 第 n 个阵元和第 1 个阵元的距离设为 $d_n^N (n=2, \dots, N)$ 。假设有 K 个不相关的目标, 发射阵列发射不同的正交编码脉冲信号, 在第 $t (t=1, \dots, J)$ 个脉冲下, 接收端匹配滤波器的输出为

$$\mathbf{x}(t) = [\mathbf{a}_r(\phi_1) \otimes \mathbf{a}_t(\theta_1), \dots, \mathbf{a}_r(\phi_K) \otimes \mathbf{a}_t(\theta_K)] \mathbf{b}(t) + \mathbf{n}(t) \quad (7.2.1)$$

其中 θ_k, ϕ_k 是分别是第 k 个目标对应于发射阵列的离开角 (Direction Of Departure, DOD) 和接收阵列的波达角 (Direction Of Arrival, DOA); $\mathbf{b}(t) = [b_1(t), b_2(t), \dots, b_K(t)]^T \in \mathbb{C}^{K \times 1}$, $b_k(t) = \beta_k e^{j2\pi f_k t / f_r}$, f_k 为多普勒频移, f_r 是脉冲重复频率, β_k 是反射幅度, 并假设在一次脉冲处理之间保持不变; $\mathbf{n}(t)$ 是 $MN \times 1$ 的高斯噪声向量, 均值为 0, 协方差矩阵为 $\sigma^2 \mathbf{I}_{MN}$; $\mathbf{a}_r(\phi_k) \otimes \mathbf{a}_t(\theta_k)$ 是对应于第 k 个目标的发射方向矢量和接收方向矢量的 Kronecker 积, 其中,

$$\mathbf{a}_r(\phi_k) = [1, \exp(-j2\pi d_2^N / \lambda \sin \phi_k), \dots, \exp(-j2\pi d_N^N / \lambda \sin \phi_k)]^T \quad (7.2.2.a)$$

$$\mathbf{a}_t(\theta_k) = [1, \exp(-j2\pi d_2^M / \lambda \sin \theta_k), \dots, \exp(-j2\pi d_M^M / \lambda \sin \theta_k)]^T \quad (7.2.2.b)$$

其中 λ 为波长。

定义 $\mathbf{A} = [\mathbf{a}_r(\phi_1) \otimes \mathbf{a}_t(\theta_1), \dots, \mathbf{a}_r(\phi_K) \otimes \mathbf{a}_t(\theta_K)]$, 则 $\mathbf{x}(t) = \mathbf{A}\mathbf{b}(t) + \mathbf{n}(t)$, 考虑对接收数据进行延迟, 如图 7.2.1 所示, 假设在延迟内, $\mathbf{a}_r(\phi_k), \mathbf{a}_t(\theta_k)$ 及反射幅度保持不变, 则

$$\begin{aligned}\mathbf{x}_p(t) &= \mathbf{A}\mathbf{b}(t + p\tau) + \mathbf{n}(t + p\tau) \\ &= \mathbf{A}\Phi_p \mathbf{b}(t) + \mathbf{n}_p(t)\end{aligned}\quad (7.2.3)$$

其中 $\Phi_p = \text{diag}(e^{j2\pi p\tau f_1/f_r}, e^{j2\pi p\tau f_2/f_r}, \dots, e^{j2\pi p\tau f_K/f_r})$ ，式中 $\text{diag}(\mathbf{v})$ 表示一个对角矩阵，其对角元素为 \mathbf{v} ； $\mathbf{n}_p(t) = \mathbf{n}(t + p\tau)$ 。

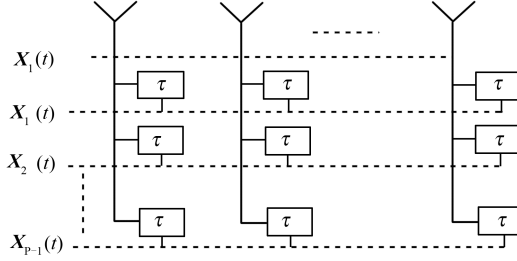


图 7.2.1 多个时延的接收信号

假设有 J 个脉冲，将 J 个脉冲下的输出、噪声以及它们的延迟写为矩阵形式：

$$\begin{aligned}\mathbf{X}_0 &= [\mathbf{x}(1), \dots, \mathbf{x}(J)] \\ \mathbf{N}_0 &= [\mathbf{n}(1), \dots, \mathbf{n}(J)] \\ \mathbf{X}_1 &= [\mathbf{x}_1(1), \dots, \mathbf{x}_1(J)] \\ \mathbf{N}_1 &= [\mathbf{n}_1(1), \dots, \mathbf{n}_1(J)] \\ &\vdots \\ \mathbf{X}_{P-1} &= [\mathbf{x}_{P-1}(1), \dots, \mathbf{x}_{P-1}(J)] \\ \mathbf{N}_{P-1} &= [\mathbf{n}_{P-1}(1), \dots, \mathbf{n}_{P-1}(J)]\end{aligned}\quad (7.2.4)$$

从中可以发现

$$\mathbf{X}_p = \begin{bmatrix} \mathbf{A}_T \mathbf{D}_1(\mathbf{A}_R) \mathbf{D}_{p+1}(\Phi) \\ \mathbf{A}_T \mathbf{D}_2(\mathbf{A}_R) \mathbf{D}_{p+1}(\Phi) \\ \vdots \\ \mathbf{A}_T \mathbf{D}_N(\mathbf{A}_R) \mathbf{D}_{p+1}(\Phi) \end{bmatrix} \mathbf{B} + \mathbf{N}_p \quad (7.2.5)$$

$$\Phi = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{j2\pi\tau f_1/f_r} & e^{j2\pi\tau f_2/f_r} & \dots & e^{j2\pi\tau f_K/f_r} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j2\pi(P-1)\tau f_1/f_r} & e^{j2\pi(P-1)\tau f_2/f_r} & \dots & e^{j2\pi(P-1)\tau f_K/f_r} \end{bmatrix} \quad (7.2.6)$$

其中 $\mathbf{D}_n(\cdot)$ 表示取矩阵的第 n 行构成对角矩阵； $\mathbf{A}_T = [\mathbf{a}_t(\theta_1), \dots, \mathbf{a}_t(\theta_K)] \in C^{M \times K}$ 为发射方向矩阵； $\mathbf{A}_R = [\mathbf{a}_r(\phi_1), \dots, \mathbf{a}_r(\phi_K)] \in C^{N \times K}$ 是接收方向矩阵， $\mathbf{B} = [\mathbf{b}(1), \dots, \mathbf{b}(J)]$ 。式 (7.2.5) 可以看成四线性模型的某一方向的切片形式。

输出数据为

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_0 \\ \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_{P-1} \end{bmatrix} = (\boldsymbol{\Phi} \odot \mathbf{A}_r \odot \mathbf{A}_t) \mathbf{B} + \mathbf{N}_x \quad (7.2.7)$$

其中 \odot 表示 Khatri-Rao 积, $\mathbf{N}_x = [\mathbf{N}_0^T, \dots, \mathbf{N}_{P-1}^T]^T$ 。

7.2.2 基于 PARAFAC 四线性分解的联合估计算法

1. 平行因子四线性模型的形成

将式 (7.2.7) 中的 $N \times P$ 个切片积累成 $M \times N \times P \times J$ 的四维数据矩阵 $\underline{\mathbf{X}}$, 则该四维矩阵有以下形式:

$$x_{n,m,p,j} = \sum_{k=1}^K b_{j,k} \psi_{p,k} a_{m,k}^t a_{n,k}^r + n_{n,m,p,j} \quad (7.2.8)$$

$$n = 1, 2, \dots, N, \quad m = 1, 2, \dots, M, \quad p = 1, \dots, P, \quad j = 1, 2, \dots, J$$

其中 $a_{m,k}^t$ 是 \mathbf{A}_t 矩阵中第 (m,k) 个元素, $a_{n,k}^r$ 是 \mathbf{A}_r 矩阵中第 (n,k) 个元素, $\psi_{p,k}$ 是矩阵 $\boldsymbol{\Phi}$ 中第 (p,k) 个元素, $b_{j,k}$ 是 \mathbf{B} 中第 (k,j) 个元素, $n_{n,m,p,j}$ 形成了四维噪声数据矩阵 $\underline{\mathbf{N}}$ 。式 (7.2.8) 称为四线性模型, 其他三种切片形式为

$$\mathbf{Y}_{mn} = \mathbf{B}^T \mathbf{D}_n(\mathbf{A}_r) \mathbf{D}_m(\mathbf{A}_t) \boldsymbol{\Phi}^T + \mathbf{N}_{mn} \quad (7.2.9)$$

$$\mathbf{Z}_{jn} = \boldsymbol{\Phi} \mathbf{D}_j(\mathbf{B}^T) \mathbf{D}_n(\mathbf{A}_r) \mathbf{A}_t^T + \mathbf{N}_{jn} \quad (7.2.10)$$

$$\mathbf{U}_{pj} = \mathbf{A}_t \mathbf{D}_p(\boldsymbol{\Phi}) \mathbf{D}_j(\mathbf{B}^T) \mathbf{A}_r^T + \mathbf{N}_{pj} \quad (7.2.11)$$

其中 $m = 1, 2, \dots, M$, $n = 1, 2, \dots, N$, $j = 1, 2, \dots, J$, $p = 1, 2, \dots, P$ 。根据式 (7.2.7), 将式 (7.2.9)、式 (7.2.10) 和式 (7.2.11) 分别表示成 $NMJ \times P$ 的矩阵 \mathbf{Y} 、 $NJP \times M$ 的矩阵 \mathbf{Z} 和 $MJP \times N$ 的矩阵 \mathbf{U} , 如式 (7.2.12)、式 (7.2.13) 和式 (7.2.14) 所示:

$$\mathbf{Y} = (\mathbf{A}_t \odot \mathbf{A}_r \odot \mathbf{B}^T) \boldsymbol{\Phi}^T + \mathbf{N}_y \quad (7.2.12)$$

$$\mathbf{Z} = (\mathbf{A}_r \odot \mathbf{B}^T \odot \boldsymbol{\Phi}) \mathbf{A}_t^T + \mathbf{N}_z \quad (7.2.13)$$

$$\mathbf{U} = (\mathbf{B}^T \odot \boldsymbol{\Phi} \odot \mathbf{A}_t) \mathbf{A}_r^T + \mathbf{N}_u \quad (7.2.14)$$

2. 平行因子四线性模型的分解

先说明平行因子四线性分解的唯一性。

定理 7.2.1^[2]: \mathbf{X} 的定义如式 (7.2.8) 所示, 如果矩阵 \mathbf{A}_r 、 \mathbf{A}_t 、 $\boldsymbol{\Phi}$ 、 \mathbf{B}^T 的 k -秩满足式 (7.2.15) 的条件, 则 \mathbf{A}_r 、 \mathbf{A}_t 、 $\boldsymbol{\Phi}$ 和 \mathbf{B}^T 对于列交换和 (复数) 尺度变换是唯一的, 即 \mathbf{A}_r 、 \mathbf{A}_t 、 $\boldsymbol{\Phi}$ 、 \mathbf{B}^T 可唯一确定。

$$k_{\mathbf{A}_r} + k_{\mathbf{A}_t} + k_{\boldsymbol{\Phi}} + k_{\mathbf{B}^T} \geq 2K + 3 \quad (7.2.15)$$

\mathbf{A}_r 、 \mathbf{A}_t 、 $\boldsymbol{\Phi}$ 都是列满秩的矩阵, 即 $k_{\mathbf{A}_r} = k_{\mathbf{A}_t} = K$, $k_{\boldsymbol{\Phi}} = \min(P, K)$, $k_{\mathbf{B}^T} = \min(J, K)$, 考虑多信源多快拍情况, 即 $K \geq 2$, $J \geq 2$, $k_{\mathbf{A}_r} + k_{\mathbf{A}_t} + k_{\boldsymbol{\Phi}} + k_{\mathbf{B}^T} \geq 2K + 3$, 满足定理 7.2.1,

\underline{X} 的 k -秩分解具有唯一性。一般来说, $J \geq K$, $P \geq K$, 如果 $J < K$, $P < K$ 则 $k_B = J$, $k_\Phi = P$, 只要 $J + P \geq 3$ 就满足定理 7.2.1, 即该算法在小样本、少延迟下也能较好的工作。

本节采用 QALS 算法对上述平行因子模型进行拟合。迭代过程中的代价函数为

$$\min_{A_r, A_t, B, \Phi} \|X - (\Phi \odot A_r \odot A_t)B\|_F \quad (7.2.16)$$

给定 A_r 、 A_t 、 Φ , 则 B 的最小二乘解为

$$\hat{B} = [(\Phi \odot A_r \odot A_t)]^\dagger X \quad (7.2.17)$$

同理, 利用式 (7.2.12), 给定 B 、 A_t 和 A_r , 则 Φ 的最小二乘解为

$$\hat{\Phi}^T = [(A_t \odot A_r \odot B^T)]^\dagger Y \quad (7.2.18)$$

利用式 (7.2.13), 给定 A_r 、 Φ 和 B , 则 A_t 的最小二乘解为

$$\hat{A}_t^T = [A_r \odot B^T \odot \Phi]^\dagger Z \quad (7.2.19)$$

利用式 (7.2.14), 给定 B 、 A_t 和 Φ , 则 A_r 的最小二乘解为

$$\hat{A}_r^T = [B^T \odot \Phi \odot A_t]^\dagger U \quad (7.2.20)$$

利用上述方法, 每次更新一个矩阵, 本次更新的矩阵立刻参与下一个矩阵的求解, 直至算法收敛。估计出的 $\hat{A}_t = A_t \Pi \Delta_1 + N_t$, $\hat{A}_r = A_r \Pi \Delta_2 + N_r$, $\hat{\Phi} = \Phi \Pi \Delta_3 + N_\Phi$, $\hat{B} = B \Pi \Delta_4 + N_B$, 其中 Π 是列模糊矩阵, Δ 为尺度模糊矩阵, 满足 $\Delta_1 \Delta_2 \Delta_3 \Delta_4 = I$, N_t 、 N_r 、 N_Φ 和 N_B 分别指所估计的误差矩阵, 它们含有目标的角度和多普勒频率信息。根据 PARAFAC 模型的唯一性定理可知, A_r 、 A_t 、 Φ 矩阵的估计值均有相同的列交换矩阵, 自然实现了角度和多普勒频率的配对。

3. 多普勒频率和角度估计

上述算法结束后, 得到时延矩阵 Φ 的估计值 $\hat{\Phi}$, 设矩阵 $\hat{\Phi}$ 的第 k 列为 $\hat{g}(f_k)$ 。对其取相角可以得到

$$\begin{aligned} \hat{h} &= \text{angle}(\hat{g}(f_k)) \\ &= [0, 2\pi\tau f_k / f_r, \dots, 2\pi(P-1)\tau f_k / f_r]^T \end{aligned} \quad (7.2.21)$$

$\text{angle}(\bullet)$ 表示取元素的相角运算, 令

$$P_1 = \begin{bmatrix} 1 & 0 \\ 1 & 2\pi\tau / f_r \\ \vdots & \vdots \\ 1 & 2\pi(P-1)\tau / f_r \end{bmatrix}, \quad b_t = \begin{bmatrix} b_0 \\ f_k \end{bmatrix}$$

则

$$P_1 b_t = \hat{h} \quad (7.2.22)$$

f_k 的最小二乘结果为

$$\begin{bmatrix} \hat{b}_0 \\ \hat{f}_k \end{bmatrix} = (P_1^T P_1)^{-1} P_1^T \hat{h} \quad (7.2.23)$$

根据式 (7.2.21) ~ 式 (7.2.23), 可依次得到 K 个多普勒频率的估计。

同样, 得到方向估计矩阵 $\hat{\mathbf{A}}_r$ 、 $\hat{\mathbf{A}}_t$, 它们的第 k 列为 $\hat{\mathbf{a}}_r(\phi_k)$ 和 $\hat{\mathbf{a}}_t(\theta_k)$, 对它们取相位角得到

$$\begin{aligned}\hat{\mathbf{g}}_1 &= -\text{angle}(\hat{\mathbf{a}}_r(\phi_k)) \\ &= [0, 2\pi d_2^N / \lambda \sin \phi_k, \dots, 2\pi d_N^N / \lambda \sin \phi_k]^T\end{aligned}\quad (7.2.24)$$

$$\begin{aligned}\hat{\mathbf{g}}_2 &= -\text{angle}(\hat{\mathbf{a}}_t(\theta_k)) \\ &= [0, 2\pi d_2^M / \lambda \sin \theta_k, \dots, 2\pi d_M^M / \lambda \sin \theta_k]^T\end{aligned}\quad (7.2.25)$$

令

$$\begin{aligned}\mathbf{P}_2 &= \begin{bmatrix} 1 & 0 \\ 1 & 2\pi d_2^N / \lambda \\ \vdots & \vdots \\ 1 & 2\pi d_N^N / \lambda \end{bmatrix}, \quad \mathbf{c}_r = \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} \\ \mathbf{P}_3 &= \begin{bmatrix} 1 & 0 \\ 1 & 2\pi d_2^M / \lambda \\ \vdots & \vdots \\ 1 & 2\pi d_M^M / \lambda \end{bmatrix}, \quad \mathbf{c}_t = \begin{bmatrix} c_3 \\ c_4 \end{bmatrix}\end{aligned}$$

则

$$\mathbf{P}_2 \mathbf{C}_r = \hat{\mathbf{g}}_1 \quad (7.2.26.a)$$

$$\mathbf{P}_3 \mathbf{C}_t = \hat{\mathbf{g}}_2 \quad (7.2.26.b)$$

\mathbf{C}_r 、 \mathbf{C}_t 的最小二乘解为

$$\begin{bmatrix} \hat{c}_0 \\ \hat{c}_1 \end{bmatrix} = (\mathbf{P}_2^T \mathbf{P}_2)^{-1} \mathbf{P}_2^T \hat{\mathbf{g}}_1 \quad (7.2.27.a)$$

$$\begin{bmatrix} \hat{c}_3 \\ \hat{c}_4 \end{bmatrix} = (\mathbf{P}_3^T \mathbf{P}_3)^{-1} \mathbf{P}_3^T \hat{\mathbf{g}}_2 \quad (7.2.27.b)$$

则 DOD 和 DOA 的估计为

$$\hat{\phi}_k = \sin^{-1}(\hat{c}_1) \quad (7.2.28.a)$$

$$\hat{\theta}_k = \sin^{-1}(\hat{c}_4) \quad (7.2.28.b)$$

至此, 可以将基于四线性分解的角度和频率联合估计算法的流程总结如下。

- ① 根据延迟的接收数据, 将 $N \times P$ 个切片积累成 $M \times N \times P \times J$ 的四维数据矩阵 \mathbf{X} 。
- ② 根据四维数据的对称性, 写成其他三种切片形式, 如式 (7.2.12) ~ 式 (7.2.14) 所示。
- ③ 根据 \mathbf{A}_r 、 \mathbf{A}_t 、 Φ 和 \mathbf{B} 的随机初值, 再根据式 (7.2.16) ~ 式 (7.2.20), 进行迭代更新, 直到算法敛。
- ④ 结合式 (7.2.23) 和式 (7.2.27), 利用最小二乘, 求出多普勒频率、DOD 和 DOA 的估计。

4. 复杂度及克拉美-罗界

本节算法的复杂度主要取决于迭代次数, 为 $O\{n[K^2(PMN+MNJ+PJN+MJP+M+N+J+P)+4MNKJP+4K^3]+2K(M+N+P)\}$, 其中 n 为迭代次数; 多维 ESPRIT 的复杂度为 $O\{M^2N^2P^2J+M^3N^3P^3+3K^3+K^2(6MPN-2MN-2MP-2NP)+K(4MNP-2MP-2NP+2)\}$; TALS 方法的复杂度为 $O\{m[K^2(PMN+MNJ+PJ+MN+J+P)+3MNKJP+3K^3]+2K(MN+P)\}$, 其中 m 为迭代次数。在快速收敛的情况下, TALS 算法的复杂度最低, 本节算法次之, 多维 ESPRIT 最高。

若 $\Omega \triangleq (\Phi \odot A_r \odot A_t)$, 根据文献[5]可以给出双基地 MIMO 雷达中的 CRB, 即

$$\text{CRB} = \frac{\sigma^2}{2J} \left\{ \text{Re} \left[\mathbf{D}^H \Pi_{\Omega}^{\perp} \mathbf{D} \oplus \hat{\mathbf{P}}_w^T \right] \right\}^{-1} \quad (7.2.29)$$

其中 $\mathbf{D} = \left[\frac{\partial \mathbf{a}_1}{\partial \theta_1}, \dots, \frac{\partial \mathbf{a}_K}{\partial \theta_K}, \frac{\partial \mathbf{a}_1}{\partial \phi_1}, \dots, \frac{\partial \mathbf{a}_K}{\partial \phi_K}, \frac{\partial \mathbf{a}_1}{\partial f_1}, \dots, \frac{\partial \mathbf{a}_K}{\partial f_K} \right]$, \mathbf{a}_k 为 Ω 的第 k 列; \oplus 代表 Hadamard 积;

$$\hat{\mathbf{P}}_w = [\hat{\mathbf{P}}_{s1}^T, \hat{\mathbf{P}}_{s1}^T, \hat{\mathbf{P}}_{s1}^T]^T, \quad \hat{\mathbf{P}}_{s1} = [\hat{\mathbf{P}}_s, \hat{\mathbf{P}}_s, \hat{\mathbf{P}}_s], \quad \hat{\mathbf{P}}_s = \frac{1}{J} \sum_{t=1}^J \mathbf{b}(t) \mathbf{b}^H(t); \quad \Pi_{\Omega}^{\perp} = \mathbf{I}_{MNP} - \Omega (\Omega^H \Omega)^{-1} \Omega^H.$$

5. 算法优点

至此, 可以将此基于四线性分解的 MIMO 雷达中的角度和多普勒频率联合估计算法的优点总结如下。

- ① 该算法是迭代算法, 不需要特征值分解或奇异值分解, 无须谱峰搜索。
- ② 不需要额外的参数配对, 可实现角度和频率的自动配对。
- ③ 无须知道反射系数, 且适用于非均匀阵列。
- ④ 算法的角度估计性能也优于多维 ESPRIT 算法^[6]和 TALS 算法^[7]。

7.2.3 仿真结果

定义求根均方误差 (Root Mean Square Error, RMSE) 为

$$\text{RMSE}_a = 1/K \sum_{k=1}^K \sqrt{1/1000 \sum_{l=1}^{1000} \left[(\hat{\phi}_{k,l} - \phi_k)^2 + (\hat{\theta}_{k,l} - \theta_k)^2 \right]} \quad (7.2.30a)$$

$$\text{RMSE}_f = 1/K \sum_{k=1}^K \sqrt{1/1000 \sum_{l=1}^{1000} \left[(\hat{f}_{k,l} - f_k)^2 \right]} \quad (7.2.30b)$$

其中 $\hat{\theta}_{k,l}$ 是 DOD 角度 θ_k 第 l 次 Monte Carlo 实验的估计值, $\hat{\phi}_{k,l}$ 是 DOA 角度 ϕ_k 第 l 次 Monte Carlo 实验的估计值, $\hat{f}_{k,l}$ 是多普勒频率 f_k 第 l 次蒙特卡罗实验的估计值。仿真中, 假设有 $K=3$ 个目标, 其角度和多普勒频率为 $(\theta_1, \phi_1, f_1) = (10^\circ, 15^\circ, 100\text{Hz})$, $(\theta_2, \phi_2, f_2) = (20^\circ, 25^\circ, 2550\text{Hz})$, $(\theta_3, \phi_3, f_3) = (30^\circ, 35^\circ, 4500\text{Hz})$, 脉冲重复频率为 $f_r = 10\text{kHz}$, 延迟级数 $P=5$, 为方便与多维 ESPRIT 方法比较, 假设阵元均匀分布, 间隔半波长。

图 7.2.2 和图 7.2.3 给出了在 $M=8$ 、 $N=6$ 、 $J=100$ 、信噪比 (Signal Noise Ratio, SNR) 分别为 5dB 和 15dB 的情况下, 该算法的角度和多普勒频率估计性能, 从图 7.2.2 和图 7.2.3 中可以看出, 目标角度和多普勒频率都可以被较准确地估计出来。

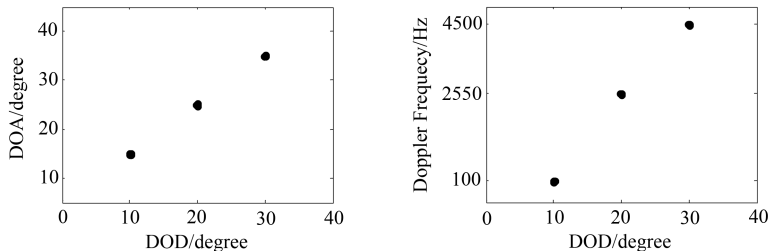


图 7.2.2 SNR=5dB 下, 该算法的估计性能

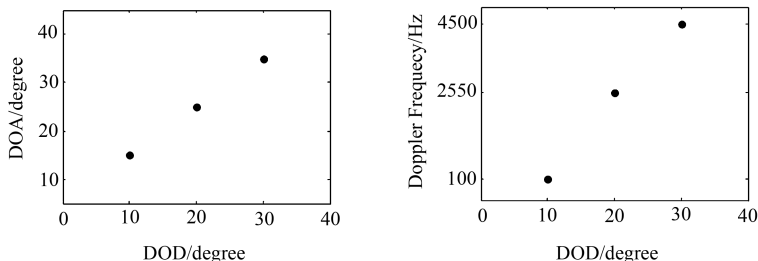


图 7.2.3 SNR=15dB 下, 该算法的估计性能

图 7.2.4 和图 7.2.5 则是在 $M=8$ 、 $N=6$ 、 $J=100$ 和 $M=6$ 、 $N=6$ 、 $J=10$ 的情况下, 该算法与多维 ESPRIT 算法^[6]、TALS 算法^[7]以及 CRB 的比较, 可以看到, 本节算法角度估计性能最优; 频率估计性能和 TALS 几乎一样, 均略优于多维 ESPRIT。

图 7.2.6 和图 7.2.7 给出了 $J=20$ 且发射和接收天线数不同的情况下, 该算法的估计性能。可以看出, 随发射和接收天线数增加, 估计性能提升。天线数增加, 分集增益增强, 性能得到提升。

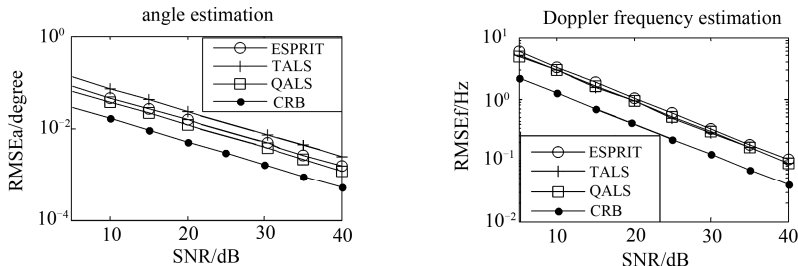


图 7.2.4 算法性能比较 ($M=8$, $N=6$, $J=100$, $K=3$)

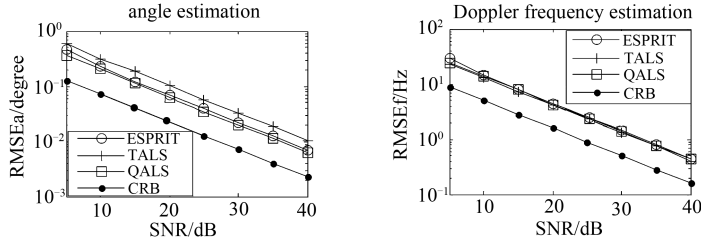


图 7.2.5 算法性能比较 ($M=6, N=6, J=10, K=3$)

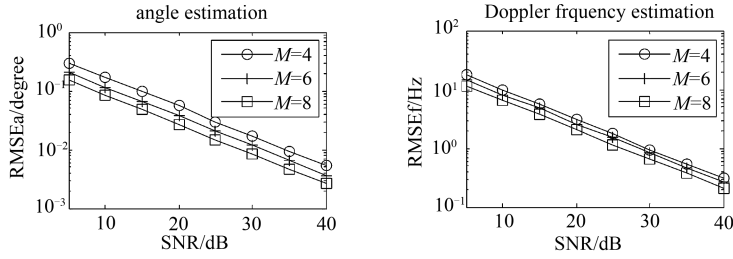


图 7.2.6 不同发射天线数下本文算法的性能 ($N=6$)

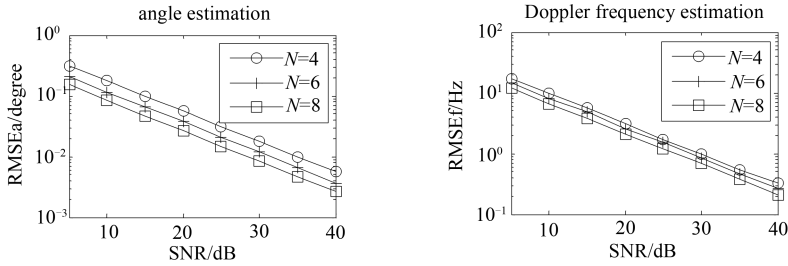


图 7.2.7 不同接收天线数下本文算法的性能 ($M=6$)

图 7.2.8 则是 $M=6, N=6$ 且快拍数不同的情况下, 本节算法的估计性能。从图 7.2.8 中可以看出, 快拍数增加, 估计性能有提升, 而且可以看到, 本节算法在低快拍下 ($J=5$), 也能准确估计角度和多普勒频率。

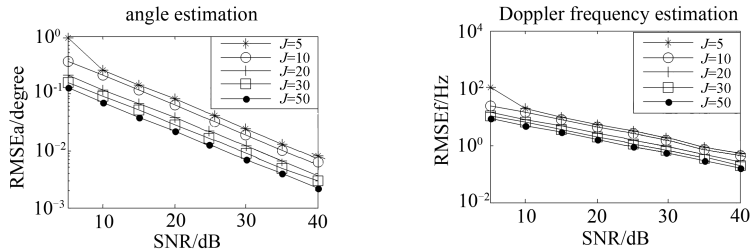


图 7.2.8 不同快拍数下该算法的性能

图 7.2.9 则是在非均匀阵下, $M=8, N=6, J=100$ 和 $\text{SNR}=5\text{dB}$ 时该算法的估计性能, 从图 7.2.9 中可以看出, 目标角度和多普勒频率都可以被较准确地估计出来, 即该算法可以工

作在非均匀阵下。

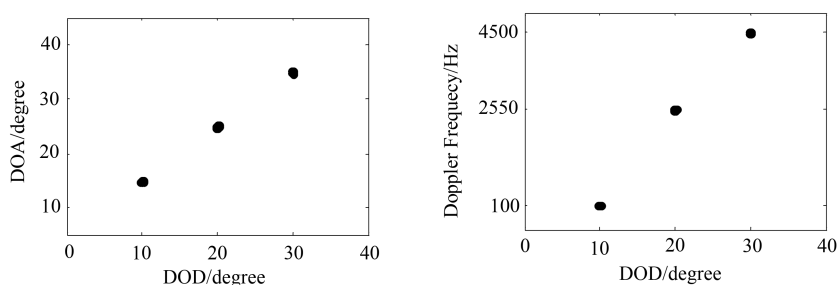


图 7.2.9 非均匀线阵, SNR=5dB 下, 该算法的估计性能

参考文献

- [1] ShiYing, Zhang Xiaofei. Quadrilinear decomposition-based blind signal detection for polarization sensitive uniform square array[J]. Progress In Electromagnetics Research, 2008, 87: 263-278.
- [2] Liu X, Sidiropoulos N D. Cramer-Rao lower bounds for low-rank decomposition of multidimensional arrays[J]. IEEE Transactions on Signal Processing, 2001, 49 (9) : 2074-2086.
- [3] 许凌云, 张小飞, 许宗泽. 平面阵列下的二维角度和频率估计[J]. 应用科学学报, 2011, 29 (2) : 187-193.
- [4] 李建峰, 张小飞. 基于四线性分解的双基地 MIMO 雷达的角度和多普勒频率联合估计 [J]. 航空学报, 2012, 33 (8) : 1474-1482.
- [5] Stoica P and Nehorai A. Performance study of conditional and unconditional direction-of-arrival estimation[J]. IEEE Transactions on Signal Processing, 1990, 38 (10) : 1783-1795.
- [6] 刘帅, 张弓, 刘文波. 基于时空结构的双基 MIMO 雷达多维参数估计[J]. 航空学报, 2010, 31 (6) : 1196-1203.
- [7] Zhang X, Xu Z, Xu L, et al. Trilinear decomposition-based transmit angle and receive angle estimation for multiple-input multiple-output radar[J]. IET Radar, Sonar & Navigation, 2011, 5 (6) : 626-631.

第8章

PARALIND 分解在通信和信号处理中的应用

本章就 PARALIND 理论在通信和信号处理中的应用给出几个例子。

8.1 非同步 CDMA 系统的 PARALIND 多用户检测

本节将异步 CDMA 系统信号检测问题和 PARALIND 联系在一起^[1]，提出确定性的盲 PARALIND 算法，其多用户检测性能非常接近非盲空时最小均方误差（Spatial-Time Minimum Mean Square Error Estimation, ST-MMSE）。本节提出的算法能解决 PARAFAC 的秩亏损问题，BER 性能比空时匹配滤波器好，且支持小样本，同时在阵列误差条件下可以工作。更重要的是，它不需要 DOA 信息和信号衰落信息。异步问题如图 8.1.1 所示。

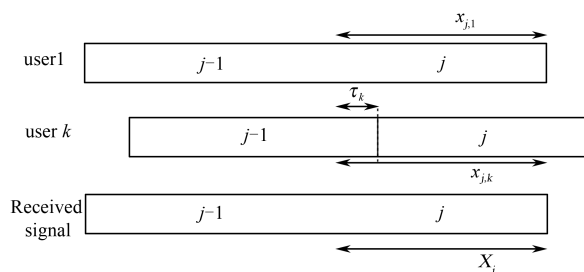


图 8.1.1 异步问题

8.1.1 数据模型

假设在 DS-CDMA 系统中有 K 个用户, 采用 BPSK 扩频调制, 每个用户都采用独立的一条路径, 所有用户异步到达天线阵列。基站采用阵元数为 M 、阵元间隔为 d 的均匀线阵, 其接收信号为

$$\mathbf{x}(t) = \sum_{k=1}^K \boldsymbol{\alpha}_k \mathbf{b}_k s_k(t - \tau_k) \boldsymbol{\beta}_k + \mathbf{n}(t) \quad (8.1.1)$$

其中 $\mathbf{n}(t)$ 为空时信道噪声, $\boldsymbol{\beta}_k$ 为第 k 个用户的信道衰落, $\boldsymbol{\alpha}_k$ 是第 k 个用户的方向向量, τ_k 为第 k 个用户的延迟, s_k 为第 k 个用户的扩频码。接收信号采用了码片速率采样。

定义 $\mathbf{h}_k = \boldsymbol{\alpha}_k \boldsymbol{\beta}_k$ 为第 k 个用户的空时信道向量。定义空时信道矩阵 $\mathbf{H} \in \mathbb{C}^{M \times K}$ 和扩频矩阵 $\mathbf{S} \in \mathbb{R}^{P \times K}$:

$$\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_K] = [\boldsymbol{\alpha}_1 \boldsymbol{\beta}_1, \boldsymbol{\alpha}_2 \boldsymbol{\beta}_2, \dots, \boldsymbol{\alpha}_K \boldsymbol{\beta}_K] = \mathbf{A} \boldsymbol{\Phi} \quad (8.1.2)$$

$$\mathbf{S} = \begin{bmatrix} s_1(1) & s_2(1) & \cdots & s_K(1) \\ s_1(2) & s_2(2) & \cdots & s_K(2) \\ \vdots & \vdots & \ddots & \vdots \\ s_1(P) & s_2(P) & \cdots & s_K(P) \end{bmatrix} \in \mathbb{R}^{P \times K} \quad (8.1.3)$$

其中 $\mathbf{A} = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \dots, \boldsymbol{\alpha}_K] \in \mathbb{C}^{M \times K}$, $[s_k(1), s_k(2), \dots, s_k(P)]$ 是第 k 个用户的扩频码, $\boldsymbol{\Phi} = \text{diag}[\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \dots, \boldsymbol{\beta}_K] \in \mathbb{C}^{K \times K}$ 。

假设第 k 个用户的延迟为 τ_k 个码片, 并且 $\tau_k < P$ (P 是扩频增益), 如图 8.1.1 所示。接收信号中第 k 个用户得到的信号分量的第 j 个符号为

$$\mathbf{x}_{j,k} = \mathbf{h}_k s_k^{(1)} \mathbf{b}_k(j-1) + \mathbf{h}_k s_k^{(2)} \mathbf{b}_k(j) \quad (8.1.4)$$

其中 $\mathbf{b}_k(j)$ 是第 k 个用户的第 j 个传输符号,

$$\mathbf{s}_k^{(1)} = [s_k(P - \tau_k + 1) \quad \cdots \quad s_k(P) \quad 0 \quad \cdots \quad 0] \in \mathbb{R}^{1 \times P} \quad (8.1.5)$$

$$\mathbf{s}_k^{(2)} = [0 \quad \cdots \quad 0 \quad s_k(1) \quad \cdots \quad s_k(P - \tau_k)] \in \mathbb{R}^{1 \times P} \quad (8.1.6)$$

其中 $[s_k(1), s_k(2), \dots, s_k(P)]$ 是第 k 个用户的扩频码。式 (8.1.4) 中的第一个部分是第 $(j-1)$ 和第 j 个符号之间的符号干扰。定义矩阵 $\mathbf{S}^{(1)} \in \mathbb{R}^{K \times P}$ 和 $\mathbf{S}^{(2)} \in \mathbb{R}^{K \times P}$

$$\mathbf{S}^{(1)} = \begin{bmatrix} \mathbf{s}_1^{(1)} \\ \mathbf{s}_2^{(1)} \\ \vdots \\ \mathbf{s}_K^{(1)} \end{bmatrix} \quad \mathbf{S}^{(2)} = \begin{bmatrix} \mathbf{s}_1^{(2)} \\ \mathbf{s}_2^{(2)} \\ \vdots \\ \mathbf{s}_K^{(2)} \end{bmatrix} \quad (8.1.7)$$

接收的无噪声信号的第 j 个符号可以表示为

$$\mathbf{X}_j = \sum_{k=1}^K \mathbf{h}_k s_k^{(1)} \mathbf{b}_k(j-1) + \sum_{k=1}^K \mathbf{h}_k s_k^{(2)} \mathbf{b}_k(j) \quad (8.1.8)$$

定义

$$\mathbf{X}_j^{(2)} = \sum_{k=1}^K \mathbf{h}_k s_k^{(2)} \mathbf{b}_k(j) \quad (8.1.9)$$

式 (8.1.9) 可以表示为

$$\mathbf{X}_j^{(2)} = \mathbf{H} \text{diag}([\mathbf{b}_1(j), \mathbf{b}_2(j), \dots, \mathbf{b}_K(j)]) \mathbf{S}^{(2)} \quad (8.1.10)$$

其中 \mathbf{H} 如式 (8.1.2) 所示。

$$\text{diag}([\mathbf{b}_1(j), \mathbf{b}_2(j), \dots, \mathbf{b}_K(j)]) = \begin{bmatrix} \mathbf{b}_1(j) & & & \\ & \mathbf{b}_2(j) & & \\ & & \ddots & \\ & & & \mathbf{b}_K(j) \end{bmatrix}$$

空时信道状态信息对于 L 个符号是常量。

$$\mathbf{X}_j^{(2)} = \mathbf{H} \mathbf{D}_j(\mathbf{B}) \mathbf{S}^{(2)}, \quad j=1, 2, \dots, L \quad (8.1.11)$$

\mathbf{B} 表达如下:

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}_1(1) & \mathbf{b}_2(1) & \dots & \mathbf{b}_K(1) \\ \mathbf{b}_1(2) & \mathbf{b}_2(2) & \dots & \mathbf{b}_K(2) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{b}_1(L) & \mathbf{b}_2(L) & \dots & \mathbf{b}_K(L) \end{bmatrix} \in \mathbb{R}^{L \times K} \quad (8.1.12)$$

其中 $\mathbf{b}_k(l)$ 是第 k 个用户的第 l 个符号。

类似地

$$\mathbf{X}_j^{(1)} = \sum_{q=1}^K \mathbf{h}_q s_q^{(1)} \mathbf{b}_q(j-1) = \mathbf{H} \mathbf{D}_j(\mathbf{B}^{(1)}) \mathbf{S}^{(1)}, \quad j=1, 2, \dots, L \quad (8.1.13)$$

其中

$$\mathbf{B}^{(1)} = \begin{bmatrix} \mathbf{b}_1(0) & \mathbf{b}_2(0) & \dots & \mathbf{b}_K(0) \\ \mathbf{b}_1(1) & \mathbf{b}_2(1) & \dots & \mathbf{b}_K(1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{b}_1(L-1) & \mathbf{b}_2(L-1) & \dots & \mathbf{b}_K(L-1) \end{bmatrix} \in \mathbb{R}^{L \times K} \quad (8.1.14)$$

接收的无噪声信号的第 j 个符号可以表示为

$$\mathbf{X}_j = \mathbf{X}_j^{(1)} + \mathbf{X}_j^{(2)} = \mathbf{H} \mathbf{D}_j(\mathbf{B}^{(1)}) \mathbf{S}^{(1)} + \mathbf{H} \mathbf{D}_j(\mathbf{B}) \mathbf{S}^{(2)}, \quad j=1, 2, \dots, L \quad (8.1.15)$$

式 (8.1.15) 也可以表示为

$$\mathbf{X}_j = [\mathbf{H} \quad \mathbf{H}] \mathbf{D}_j(\mathbf{B}_E) \begin{bmatrix} \mathbf{S}^{(1)} \\ \mathbf{S}^{(2)} \end{bmatrix} = \mathbf{H} \mathbf{\Gamma} \mathbf{D}_j(\mathbf{B}_E) \mathbf{S}_E^T, \quad j=1, 2, \dots, L \quad (8.1.16)$$

其中

$$\mathbf{B}_E = [\mathbf{B}^{(1)} \quad \mathbf{B}] \in \mathbb{R}^{L \times 2K} \quad (8.1.17)$$

$$\mathbf{S}_E = \begin{bmatrix} \mathbf{S}^{(1)} \\ \mathbf{S}^{(2)} \end{bmatrix}^T \in \mathbb{R}^{P \times 2K} \quad (8.1.18)$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} 1 & & & 1 & & \\ & 1 & & & 1 & \\ & & \ddots & & & \ddots \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix} \in \mathbb{R}^{K \times 2K} \quad (8.1.19)$$

其中 $\boldsymbol{\Gamma}$ 是相关矩阵, 接收信号可以表示为 PARALIND 模型, \mathbf{X}_j 为时间方向的第 j 个切片。

根据式 (8.1.16), 接收的无噪声信号为

$$\begin{aligned} \mathbf{X} &= [\mathbf{X}_1 \quad \mathbf{X}_2 \quad \cdots \quad \mathbf{X}_L] \\ &= \mathbf{H}\boldsymbol{\Gamma} \begin{bmatrix} \mathbf{D}_1(\mathbf{B}_E)\mathbf{S}_E^\top & \mathbf{D}_2(\mathbf{B}_E)\mathbf{S}_E^\top & \cdots & \mathbf{D}_L(\mathbf{B}_E)\mathbf{S}_E^\top \end{bmatrix} \\ &= \mathbf{H}\boldsymbol{\Gamma}(\mathbf{B}_E \odot \mathbf{S}_E)^\top \end{aligned} \quad (8.1.20)$$

当延迟大于一个符号时间, 即 $\tau_k \geq P$ (P 为扩频增益) 时, 例如, 当 $mP \leq \tau_k \leq (m+1)P$, m 为正整数时, 式 (8.1.4) 变为

$$\mathbf{x}_{j,k} = \mathbf{h}_k \mathbf{s}_k^{(1)} \mathbf{b}_k(j-m-1) + \mathbf{h}_k \mathbf{s}_k^{(2)} \mathbf{b}_k(j-m) \quad (8.1.21)$$

其中

$$\mathbf{s}_k^{(1)} = [\mathbf{s}_k(P-\tau_k+mP+1) \quad \cdots \quad \mathbf{s}_k(P) \quad 0 \quad \cdots \quad 0] \in \mathbb{R}^{k \times P} \quad (8.1.22)$$

$$\mathbf{s}_k^{(2)} = [0 \quad \cdots \quad 0 \quad \mathbf{s}_k(1) \quad \cdots \quad \mathbf{s}_k(P-\tau_k+mP)] \in \mathbb{R}^{k \times P} \quad (8.1.23)$$

通过相同的分析, 接收信号也可以建模为 PARALIND。

8.1.2 异步 CDMA 系统中基于 PARALIND 的盲空时多用户检测

在线性相关情况下, PARAFAC 模型出现秩亏。PARAFAC-ALS 算法对于三线性模型可以收敛到全局最小值, 在存在相关的情况下算法失效。在这里使用 PARALIND 来解决这个问题。式 (8.1.16) 中的信号建模为 PARALIND。

1. PARALIND 分解

PARALIND 模型的代价函数为

$$\min \left\| \tilde{\mathbf{X}} - \mathbf{H}\boldsymbol{\Gamma}(\mathbf{B}_E \odot \mathbf{S}_E)^\top \right\|_F^2 \quad (8.1.24)$$

其中 $\tilde{\mathbf{X}}$ 为接收含噪信号。 $\|\cdot\|_F$ 为 F 范数。

根据式 (8.1.16) 可得

$$\text{vec}(\mathbf{X}_j) = [\mathbf{S}_E \mathbf{D}_j(\mathbf{B}_E) \otimes \mathbf{H}] \text{vec}(\boldsymbol{\Gamma}) \quad (8.1.25)$$

其中 $\text{vec}(\mathbf{X})$ 由 \mathbf{X} 沿列方向排成一系列得到。考虑所有切片同时进行可以得到

$$\begin{bmatrix} \text{vec}(\mathbf{X}_1) \\ \text{vec}(\mathbf{X}_2) \\ \vdots \\ \text{vec}(\mathbf{X}_L) \end{bmatrix} = \begin{bmatrix} \mathbf{S}_E \mathbf{D}_1(\mathbf{B}_E) \otimes \mathbf{H} \\ \mathbf{S}_E \mathbf{D}_2(\mathbf{B}_E) \otimes \mathbf{H} \\ \vdots \\ \mathbf{S}_E \mathbf{D}_L(\mathbf{B}_E) \otimes \mathbf{H} \end{bmatrix} \text{vec}(\boldsymbol{\Gamma}) \quad (8.1.26)$$

式 (8.1.26) 可以表示为

$$\text{vec}(\mathbf{X}) = [(\mathbf{B}_E \odot \mathbf{S}_E) \otimes \mathbf{H}] \text{vec}(\mathbf{\Gamma}) \quad (8.1.27)$$

相关矩阵 $\mathbf{\Gamma}$ 为

$$\text{vec}(\mathbf{\Gamma}) = [(\mathbf{S}_E^T \mathbf{S}_E) \oplus (\mathbf{B}_E^T \mathbf{B}_E) \otimes (\mathbf{H}^H \mathbf{H})]^{-1} \text{vec}(\sum_{j=1}^L \mathbf{H}^H \mathbf{X}_j \mathbf{S}_E \mathbf{D}_j(\mathbf{B}_E)) \quad (8.1.28)$$

其中 \oplus 代表 Hadamard 积。

相关矩阵 $\mathbf{\Gamma}$ 也要考虑到

$$(\mathbf{B}_E \odot \mathbf{S}_E)^T (\mathbf{B}_E \odot \mathbf{S}_E) = (\mathbf{S}_E^T \mathbf{S}_E) \oplus (\mathbf{B}_E^T \mathbf{B}_E)$$

同时, 更新矩阵 \mathbf{H} , \mathbf{S}_E , \mathbf{B}_E 为

$$\mathbf{H} = (\sum_{j=1}^L \mathbf{X}_j \mathbf{S}_E \mathbf{D}_j(\mathbf{B}_E) \mathbf{\Gamma}^T) \{ \mathbf{\Gamma} [(\mathbf{S}_E^T \mathbf{S}_E) \oplus (\mathbf{B}_E^T \mathbf{B}_E)] \mathbf{\Gamma}^T \}^{-1} \quad (8.1.29)$$

$$\mathbf{S}_E = (\sum_{j=1}^L \mathbf{X}_j^H \mathbf{H} \mathbf{\Gamma} \mathbf{D}_j(\mathbf{B}_E)) \{ (\mathbf{\Gamma}^T \mathbf{H}^H \mathbf{H} \mathbf{\Gamma}) \oplus (\mathbf{B}_E^T \mathbf{B}_E) \}^{-1} \quad (8.1.30)$$

$$\text{diag}^{-1}(\mathbf{D}_j(\mathbf{B}_E)) = \{ (\mathbf{S}_E^T \mathbf{S}_E) \oplus (\mathbf{\Gamma}^T \mathbf{H}^H \mathbf{H} \mathbf{\Gamma}) \}^{-1} \text{diag}(\mathbf{H} \mathbf{\Gamma}^T \mathbf{X}_j \mathbf{S}_E), \quad j=1, 2, \dots, L \quad (8.1.31)$$

其中运算符 $\text{diag}^{-1}(\cdot)$ 表示取对角线元素为一个列向量。至此, 已经提出了在异步 CDMA 系统中基于 PARALIND 的空时多用户检测算法, 步骤如下:

- ① 初始化矩阵 \mathbf{H} 和 \mathbf{B}_E , \mathbf{S}_E 由扩频码和延迟信息得到。 $\mathbf{\Gamma}$ 如式 (8.1.19) 所示。
- ② 根据式 (8.1.29) 更新矩阵 \mathbf{H} 。
- ③ 根据式 (8.1.31) 更新矩阵 \mathbf{B}_E 。
- ④ 重复步骤 2 和 3 直到收敛。

2. PARALIND 分解唯一性

根据式 (8.1.16), 两个切片可以表示为

$$\mathbf{X}_1 = \bar{\mathbf{H}} \bar{\mathbf{S}}_E^T, \quad \mathbf{X}_2 = \bar{\mathbf{H}} \mathbf{D} \bar{\mathbf{S}}_E^T \quad (8.1.32)$$

其中 $\bar{\mathbf{H}} = \mathbf{H} \mathbf{\Gamma}$, $\bar{\mathbf{S}}_E = \mathbf{S}_E \mathbf{D}_1(\mathbf{B}_E)$, $\mathbf{D} = \mathbf{D}_2(\mathbf{B}_E) \mathbf{D}_1(\mathbf{B}_E)^{-1}$, 而

$$\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{H}} \\ \bar{\mathbf{H}} \mathbf{D} \end{bmatrix} \bar{\mathbf{S}}_E^T \quad (8.1.33)$$

$\bar{\mathbf{S}}_E$ 为列满秩矩阵。假设 $\text{span}(\mathbf{U}) = \text{span}(\begin{bmatrix} \bar{\mathbf{H}} \\ \bar{\mathbf{H}} \mathbf{D} \end{bmatrix})$ 。因此存在一个非奇异矩阵 \mathbf{T} 满足

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{H}} \mathbf{T} \\ \bar{\mathbf{H}} \mathbf{D} \mathbf{T} \end{bmatrix} \quad (8.1.34)$$

构造自相关矩阵和互相关矩阵

$$\mathbf{R}_1 = \mathbf{U}_1^H \mathbf{U}_1 = \mathbf{T}^H \bar{\mathbf{H}}^H \bar{\mathbf{H}} \mathbf{T} = \mathbf{G} \mathbf{T} \quad (8.1.35)$$

$$\mathbf{R}_2 = \mathbf{U}_1^H \mathbf{U}_2 = \mathbf{T}^H \bar{\mathbf{H}}^H \bar{\mathbf{H}} \mathbf{D} \mathbf{T} = \mathbf{G} \mathbf{D} \mathbf{T} \quad (8.1.36)$$

其中 $\mathbf{G} = \mathbf{T}^H \bar{\mathbf{H}}^H \bar{\mathbf{H}}$ 。

矩阵 \mathbf{R}_1 和 \mathbf{R}_2 为满秩方阵。它们满足

$$(\mathbf{R}_2 \mathbf{R}_1^{-1})^T (\mathbf{G}^{-1})^T = (\mathbf{G}^{-1})^T \mathbf{D} \quad (8.1.37)$$

因此 \mathbf{D} 和 $(\mathbf{G}^{-1})^T$ 为矩阵 $(\mathbf{R}_2 \mathbf{R}_1^{-1})^T$ 的特征值和其对应的特征向量。特征值和特征向量是一一对应的。 \mathbf{D} 是唯一的。由 $\mathbf{R}_1 = \mathbf{G}\mathbf{T}$ 得到 $\mathbf{T} = \mathbf{G}^{-1}\mathbf{R}_1$ ，由 $\mathbf{U}_1 = \bar{\mathbf{H}}\mathbf{T}$ 得到 $\bar{\mathbf{H}} = \mathbf{U}_1\mathbf{T}^{-1}$ ，由 $\mathbf{X}_1 = \bar{\mathbf{H}}\bar{\mathbf{S}}_E^T$ 得到 $\bar{\mathbf{S}}_E^T = \bar{\mathbf{H}}^+ \mathbf{X}_1$ 。另外，尺度模糊和列模糊存在。前者可以通过嵌入信息解决，后者可以利用已知的 \mathbf{S}_E 解决。

8.1.3 仿真结果

令 $\tilde{\mathbf{X}}_j = \mathbf{H}\mathbf{F}\mathbf{D}_j(\mathbf{B}_E)\mathbf{S}_E^T + \mathbf{N}_j$, $j=1,2,\dots,L$ 为接收的含噪信号，其中 \mathbf{N}_j 为高斯白噪声矩阵。定义 SNR:

$$\text{SNR} = 10 \log_{10} \frac{\sum_{j=1}^L \|\mathbf{H}\mathbf{F}\mathbf{D}_j(\mathbf{B}_E)\mathbf{S}_E^T\|_F^2}{\sum_{j=1}^L \|\mathbf{N}_j\|_F^2} \text{dB} \quad (8.1.38)$$

利用 Monte Carlo 仿真来评估本节算法的误比特率 (Bit Error Ratio, BER)，仿真的次数为 2000。对比 PARALIND 算法与非盲空时最小均方误差和空时匹配滤波器 (Spatial Time-Matched Filter, ST-MF)，ST-MMSE 提供了盲算法的性能界。对比盲 PARALIND 算法，非盲 ST-MMSE 需要 DOA、信噪比 SNR 和信道衰落信息，而 PARALIND 不需要 DOA 信息和信道衰落信息。

基站使用天线数为 6，天线间距为半波长的线阵，扩频增益为 16，假设采用 BPSK 调制信号和加性高斯白噪声。注意， L 为快拍数， K 为用户数。

仿真 1: 图 8.1.2~图 8.1.5 表示在不同 L 下这些算法的性能，用户数为 3。图 8.1.2 表示大样本情况，即 $L=200$ 下的仿真结果。从图 8.1.2 可以发现，盲 PARALIND 算法非常接近非盲 ST-MMSE，并且 PARALIND 算法比 ST-MF 有更好的性能。图 8.1.3 和图 8.1.4 描述了在 $L=100$ 和 $L=50$ 时的仿真结果。从图 8.1.2~图 8.1.4 可以发现，随着 L 的下降，盲 PARALIND 和非盲的 ST-MMSE 接收器性能差距增加。图 8.1.5 显示了在小样本的情况 ($L=20$) 下的仿真结果，显示了 PARALIND 算法在小样本情况下依旧性能良好，性能比 ST-MF 接收器好。

仿真 2: 本仿真描述了 PARALIND 算法在不同用户数 K 情况下的算法性能。图 8.1.6 表示了 $L=100$ 和不同 K 下的结果，显示了 PARALIND 算法性能随着用户数的增加而下降。

仿真 3: 阵列误差向量为 $\mathbf{g} = [1.0000, 0.6071 - 0.6953i, 1.0083 + 0.9059i, 0.3497 - 0.7167i, 0.9693 + 1.2916i, 0.5343 + 0.3883i]$ 。假设对 $\text{DOA}=\theta$ 的阵列响应为 $\mathbf{a}(\theta)$ ，因此带有阵列误差的阵列响应向量为 $\text{diag}(\mathbf{g})\mathbf{a}(\theta)$ 。采样数 L 为 100，用户数为 3。图 8.1.7 显示了在阵列误差情况下的盲 PARALIND 算法性能，说明图 8.1.7 表明在阵列误差情况下，盲 PARALIND 算法也有很好的性能，说明盲 PARALIND 算法对于阵列误差有很好的鲁棒性。

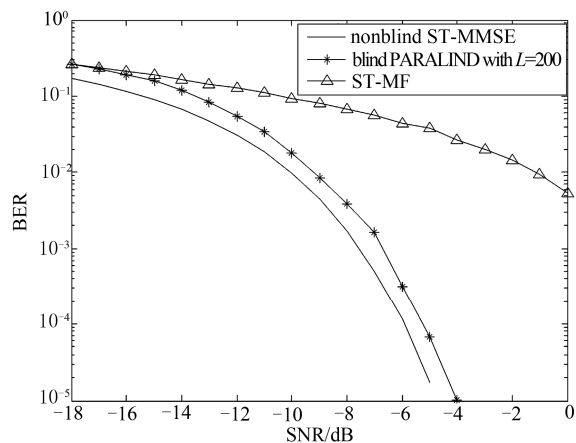


图 8.1.2 $K=3$ 和 $L=200$ 时的算法性能对比

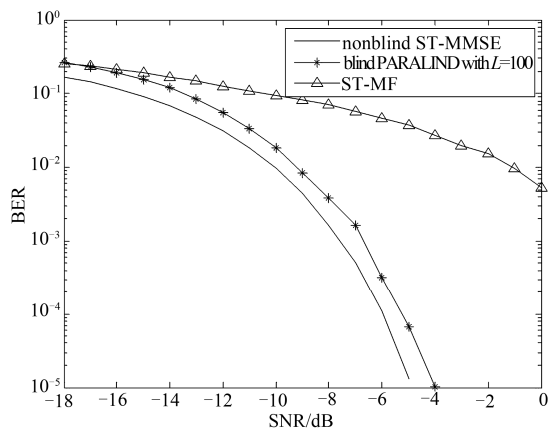


图 8.1.3 $K=3$ 和 $L=100$ 时的算法性能对比

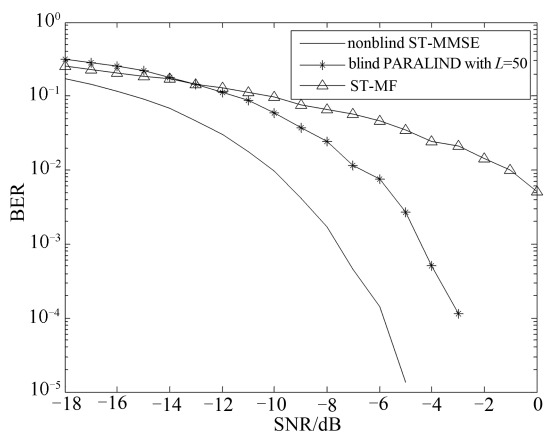


图 8.1.4 $K=3$ 和 $L=50$ 时的算法性能对比

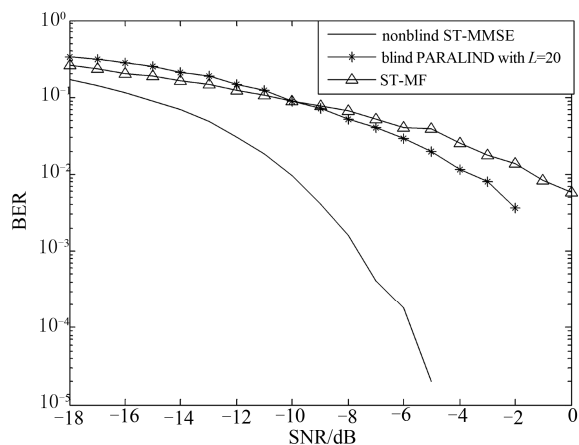
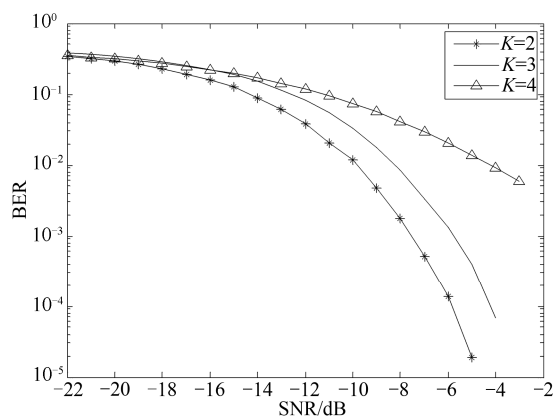
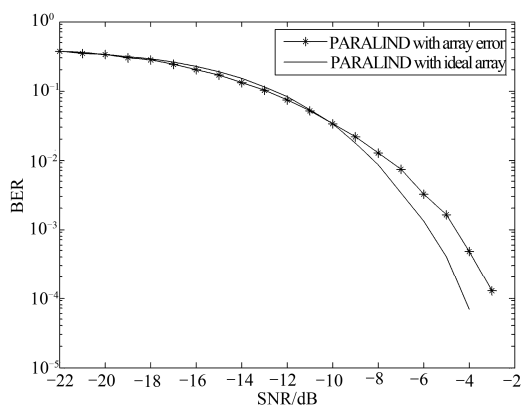
图 8.1.5 $K=3$ 和 $L=20$ 时的算法性能对比图 8.1.6 不同 K 情况下的算法性能对比

图 8.1.7 阵列误差下的算法性能

8.2 多径下 CDMA 系统的 PARALIND 多用户检测

本节将多径 CDMA 系统信号检测问题和 PARALIND 联系在一起^[2], 并提出多径 CDMA 系统中的 PARALIND 盲多用户检测算法。

8.2.1 数据模型

假设在 DS-CDMA 系统中有 K 个用户, 使用 BPSK 扩频调制, 所有用户经过多条路径到达接收机, 基站采用 M 个阵元的均匀圆阵。接收信号为

$$\mathbf{x}(t) = \sum_{k=1}^K \sum_{l=1}^{r_k} \mathbf{a}_{k,l} b_k(t) s_k(t - \tau_{k,l}) \beta_{k,l} + \mathbf{n}(t) \quad (8.2.1)$$

其中 $\mathbf{n}(t)$ 是空时信道噪声, $\beta_{k,l}$ 是第 k 个用户的第 l 条路径的信道衰落, $\mathbf{a}_{k,l}$ 是第 k 个用户的第 l 条路径的方向向量, $\tau_{k,l}$ 为第 k 个用户第 l 条路径的延迟, s_k 是第 k 个用户的扩频编码, $b_k(t)$ 是第 k 个用户的传输信号, r_k 是第 k 个用户的多径数。定义 $\mathbf{h}_{k,l} = \mathbf{a}_{k,l} \beta_{k,l}$ 是第 k 个用户第 l 条路径的空时信道向量。接收信号采用码片速率采样。

假设第 k 个用户第 l 条路径上的延迟为 $\tau_{k,l}$ 码片, 并且 $\tau_{k,l} < P$ (P 是扩频增益), 如图 8.2.1 所示。第 k 个用户的接收信号的第 j 个符号中的第 l 条路径分量可以表示为

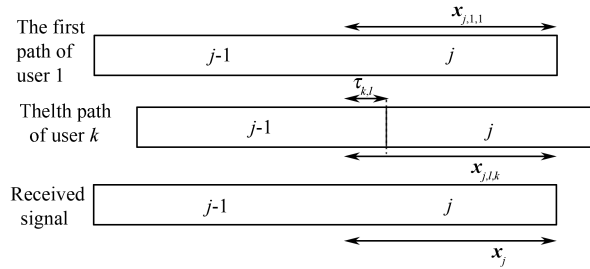


图 8.2.1 异步问题

$$\mathbf{x}_{j,l,k} = \mathbf{h}_{k,l} \mathbf{s}_{k,l}^{(1)} b_k(j-1) + \mathbf{h}_{k,l} \mathbf{s}_{k,l}^{(2)} b_k(j) \quad (8.2.2)$$

其中 $b_k(j)$ 是第 k 个用户的第 j 个传输符号。

$$\mathbf{s}_{k,l}^{(1)} = [s_k(P - \tau_{k,l} + 1) \cdots s_k(P) \ 0 \cdots 0] \in \mathbb{R}^{1 \times P} \quad (8.2.3)$$

$$\mathbf{s}_{k,l}^{(2)} = [0 \cdots 0 \ s_k(1) \cdots s_k(P - \tau_{k,l})] \in \mathbb{R}^{1 \times P} \quad (8.2.4)$$

式中, $[s_k(1), s_k(2), \dots, s_k(P)]$ 是第 k 个用户的扩频码。式 (8.2.2) 中的第一部分可以当做第 $(j-1)$ 和第 j 个符号之间的干扰。

定义 $\mathbf{r} = \sum_{k=1}^K \mathbf{r}_k$ 。接收无噪声信号的第 j 个符号可以表示为

$$\mathbf{X}_j = \sum_{k=1}^K \sum_{l=1}^{r_k} \mathbf{x}_{j,l,k} = \sum_{k=1}^K \sum_{l=1}^{r_k} \mathbf{h}_{k,l} \mathbf{s}_{k,l}^{(1)} b_k(j-1) + \sum_{k=1}^K \sum_{l=1}^{r_k} \mathbf{h}_{k,l} \mathbf{s}_{k,l}^{(2)} b_k(j) \quad (8.2.5)$$

定义

$$\mathbf{X}_j^{(2)} = \sum_{k=1}^K \sum_{l=1}^{r_k} \mathbf{h}_{k,l} \mathbf{s}_{k,l}^{(2)} b_k(j) = \mathbf{H} \text{diag}(\mathbf{b}_j^{(2)}) \mathbf{S}^{(2)} \quad (8.2.6)$$

其中

$$\mathbf{H} = [\mathbf{h}_{1,1}, \dots, \mathbf{h}_{1,r_1}, \mathbf{h}_{2,1}, \dots, \mathbf{h}_{2,r_2}, \dots, \mathbf{h}_{K,1}, \dots, \mathbf{h}_{K,r_K}] \in \mathbb{C}^{M \times r} \quad (8.2.7)$$

$$\mathbf{S}^{(2)} = [\mathbf{s}_{1,1}^{(2)\text{T}}, \dots, \mathbf{s}_{1,r_1}^{(2)\text{T}}, \mathbf{s}_{2,1}^{(2)\text{T}}, \dots, \mathbf{s}_{2,r_2}^{(2)\text{T}}, \dots, \mathbf{s}_{K,1}^{(2)\text{T}}, \dots, \mathbf{s}_{K,r_K}^{(2)\text{T}}]^{\text{T}} \in \mathbb{R}^{r \times P} \quad (8.2.8)$$

$$\mathbf{b}_j^{(2)} = \left[\underbrace{b_1(j), \dots, b_1(j)}_{r_1}, \dots, \underbrace{b_K(j), \dots, b_K(j)}_{r_K} \right] \in \mathbb{R}^{1 \times r} \quad (8.2.9)$$

假设 J 个符号空时信道状态信息是常量, 式 (8.2.6) 可以表示为

$$\mathbf{X}_j^{(2)} = \mathbf{H} \mathbf{D}_j(\mathbf{B}^{(2)}) \mathbf{S}^{(2)}, \quad j=1, 2, \dots, J \quad (8.2.10)$$

$\mathbf{B}^{(2)}$ 表示为

$$\mathbf{B}^{(2)} = [\mathbf{b}_1^{(2)\text{T}}, \mathbf{b}_2^{(2)\text{T}}, \dots, \mathbf{b}_J^{(2)\text{T}}]^{\text{T}} \in \mathbb{R}^{J \times r} \quad (8.2.11)$$

类似地, 定义

$$\mathbf{X}_j^{(1)} = \sum_{q=1}^K \mathbf{h}_q \mathbf{s}_q^{(1)} b_q(j-1) = \mathbf{H} \mathbf{D}_j(\mathbf{B}^{(1)}) \mathbf{S}^{(1)}, \quad j=1, 2, \dots, J \quad (8.2.12)$$

其中

$$\mathbf{S}^{(1)} = [\mathbf{s}_{1,1}^{(1)\text{T}}, \dots, \mathbf{s}_{1,r_1}^{(1)\text{T}}, \mathbf{s}_{2,1}^{(1)\text{T}}, \dots, \mathbf{s}_{2,r_2}^{(1)\text{T}}, \dots, \mathbf{s}_{K,1}^{(1)\text{T}}, \dots, \mathbf{s}_{K,r_K}^{(1)\text{T}}]^{\text{T}} \in \mathbb{R}^{r \times P} \quad (8.2.13)$$

$$\mathbf{B}^{(1)} = [\mathbf{b}_1^{(1)\text{T}}, \mathbf{b}_2^{(1)\text{T}}, \dots, \mathbf{b}_J^{(1)\text{T}}]^{\text{T}} \in \mathbb{R}^{J \times r} \quad (8.2.14)$$

其中

$$\mathbf{b}_j^{(1)} = \left[\underbrace{b_1(j-1), \dots, b_1(j-1)}_{r_1}, \dots, \underbrace{b_K(j-1), \dots, b_K(j-1)}_{r_K} \right] \in \mathbb{R}^{1 \times r} \quad (8.2.15)$$

接收无噪声信号的第 j 个符号为

$$\mathbf{X}_j = \mathbf{X}_j^{(1)} + \mathbf{X}_j^{(2)} = \mathbf{H} \mathbf{D}_j(\mathbf{B}^{(1)}) \mathbf{S}^{(1)} + \mathbf{H} \mathbf{D}_j(\mathbf{B}^{(2)}) \mathbf{S}^{(2)}, \quad j=1, 2, \dots, J \quad (8.2.16)$$

式 (8.2.16) 表示为

$$\mathbf{X}_j = [\mathbf{H} \quad \mathbf{H}] \mathbf{D}_j(\mathbf{B}_E) \begin{bmatrix} \mathbf{S}^{(1)} \\ \mathbf{S}^{(2)} \end{bmatrix} = \mathbf{H} \mathbf{D}_j(\mathbf{B}_E) \mathbf{S}_E^{\text{T}}, \quad j=1, 2, \dots, J \quad (8.2.17)$$

其中

$$\mathbf{S}_E = \begin{bmatrix} \mathbf{S}^{(1)} \\ \mathbf{S}^{(2)} \end{bmatrix}^{\text{T}} \in \mathbb{R}^{P \times 2r} \quad (8.2.18)$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} 1 & & & 1 & & \\ & 1 & & & 1 & \\ & & \ddots & & & \ddots \\ & & & 1 & & 1 \end{bmatrix} \in \mathbb{R}^{r \times 2r} \quad (8.2.19)$$

$$\boldsymbol{B}_E = [\boldsymbol{B}^{(1)} \quad \boldsymbol{B}^{(2)}] = [\boldsymbol{B}_1, \boldsymbol{B}_2] \boldsymbol{T} = \boldsymbol{B} \boldsymbol{T} \in \mathbb{R}^{J \times 2r} \quad (8.2.20)$$

其中

$$\boldsymbol{T} = \begin{bmatrix} \bar{\boldsymbol{T}} & \\ & \bar{\boldsymbol{T}} \end{bmatrix} \in \mathbb{R}^{2K \times 2r}, \quad \bar{\boldsymbol{T}} = \begin{bmatrix} \mathbf{1}_{r_1} & 0 \\ & \ddots \\ 0 & \mathbf{1}_{r_K} \end{bmatrix} \in \mathbb{R}^{K \times r},$$

$\mathbf{1}_{r_i}$ 是 $1 \times r_i$ 向量。

$$\boldsymbol{B}_1 = \begin{bmatrix} b_1(0) & b_2(0) & \cdots & b_K(0) \\ b_1(1) & b_2(1) & \cdots & b_K(1) \\ \vdots & \vdots & \ddots & \vdots \\ b_1(J-1) & b_2(J-1) & \cdots & b_K(J-1) \end{bmatrix} \in \mathbb{R}^{J \times K} \quad (8.2.21)$$

$$\boldsymbol{B}_2 = \begin{bmatrix} b_1(1) & b_2(1) & \cdots & b_K(1) \\ b_1(2) & b_2(2) & \cdots & b_K(2) \\ \vdots & \vdots & \ddots & \vdots \\ b_1(J) & b_2(J) & \cdots & b_K(J) \end{bmatrix} \in \mathbb{R}^{J \times K} \quad (8.2.22)$$

$$\boldsymbol{B} = [\boldsymbol{B}_1, \boldsymbol{B}_2] \in \mathbb{R}^{J \times 2K} \quad (8.2.23)$$

式 (8.2.17) 也可以表示为

$$\boldsymbol{X}_j = \boldsymbol{H} \boldsymbol{\Gamma} \boldsymbol{D}_j (\boldsymbol{B} \boldsymbol{T}) \boldsymbol{S}_E^T, \quad j=1,2,\cdots,J \quad (8.2.24)$$

其中 $\boldsymbol{\Gamma}$ 和 \boldsymbol{T} 为相关矩阵, 接收信号可以表示 PARALIND, \boldsymbol{X}_j 为时间方向的第 j 个切片。

根据式 (8.2.24) 中的三线性模型, 式 (8.2.24) 可以通过重置表示为

$$\begin{aligned} \boldsymbol{Y}_p' &= \boldsymbol{B} \boldsymbol{T} \boldsymbol{D}_p (\boldsymbol{S}_E) (\boldsymbol{H} \boldsymbol{\Gamma})^T, \quad p=1,2,\cdots,P; \quad \boldsymbol{Z}_m' = \boldsymbol{S}_E \boldsymbol{D}_m (\boldsymbol{H} \boldsymbol{\Gamma}) (\boldsymbol{B} \boldsymbol{T})^T, \quad m=1,2,\cdots,M \\ \boldsymbol{Y}_p &= [\boldsymbol{Y}_p']^T = \boldsymbol{H} \boldsymbol{\Gamma} \boldsymbol{D}_p (\boldsymbol{S}_E) \boldsymbol{T}^T \boldsymbol{B}^T, \quad p=1,2,\cdots,P \end{aligned} \quad (8.2.25)$$

由式 (8.2.25) 可得接收无噪声信号

$$\begin{aligned} \boldsymbol{Y} &= [\boldsymbol{Y}_1 \quad \boldsymbol{Y}_2 \quad \cdots \quad \boldsymbol{Y}_P] \\ &= \boldsymbol{H} \boldsymbol{\Gamma} \begin{bmatrix} \boldsymbol{D}_1 (\boldsymbol{S}_E) \boldsymbol{T}^T \boldsymbol{B}^T & \boldsymbol{D}_2 (\boldsymbol{S}_E) \boldsymbol{T}^T \boldsymbol{B}^T & \cdots & \boldsymbol{D}_P (\boldsymbol{S}_E) \boldsymbol{T}^T \boldsymbol{B}^T \end{bmatrix} \\ &= \boldsymbol{H} \boldsymbol{\Gamma} (\boldsymbol{S}_E \odot \boldsymbol{B} \boldsymbol{T})^T \end{aligned} \quad (8.2.26)$$

8.2.2 盲 PARALIND 多用户检测

在这里使用 PARALIND 来解决这个问题。式 (8.2.16) 中的信号建模为 PARALIND, PARALIND 模型的代价函数为

$$\min \|\tilde{\mathbf{Y}} - \mathbf{H}\mathbf{\Gamma}(\mathbf{S}_E \odot \mathbf{B}\mathbf{T})^T\|_F^2 \quad (8.2.27)$$

其中 $\tilde{\mathbf{Y}}$ 为接收含噪信号, $\|\cdot\|_F$ 为 F 范数。

根据式 (8.2.25)

$$\text{vec}(\mathbf{Y}_p) = \text{vec}(\mathbf{H}\mathbf{\Gamma}\mathbf{D}_p(\mathbf{S}_E)\mathbf{T}^T\mathbf{B}^T) = [\mathbf{B}\mathbf{T}\mathbf{D}_p(\mathbf{S}_E) \otimes \mathbf{H}]\text{vec}(\mathbf{\Gamma}) \quad (8.2.28)$$

考虑所有切片, 得到

$$\begin{bmatrix} \text{vec}(\mathbf{Y}_1) \\ \text{vec}(\mathbf{Y}_2) \\ \vdots \\ \text{vec}(\mathbf{Y}_P) \end{bmatrix} = \begin{bmatrix} \mathbf{B}\mathbf{T}\mathbf{D}_1(\mathbf{S}_E) \otimes \mathbf{H} \\ \mathbf{B}\mathbf{T}\mathbf{D}_2(\mathbf{S}_E) \otimes \mathbf{H} \\ \vdots \\ \mathbf{B}\mathbf{T}\mathbf{D}_P(\mathbf{S}_E) \otimes \mathbf{H} \end{bmatrix} \text{vec}(\mathbf{\Gamma}) \quad (8.2.29)$$

式 (8.2.29) 可以表示为

$$\text{vec}(\mathbf{Y}) = [(\mathbf{S}_E \odot \mathbf{B}\mathbf{T}) \otimes \mathbf{H}]\text{vec}(\mathbf{\Gamma}) \quad (8.2.30)$$

更新 $\mathbf{\Gamma}$ 为

$$\text{vec}(\mathbf{\Gamma}) = [(\mathbf{T}^T\mathbf{B}^T\mathbf{B}\mathbf{T}) \oplus (\mathbf{S}_E^T\mathbf{S}_E) \otimes (\mathbf{H}^T\mathbf{H})]^{-1} \text{vec}(\sum_{p=1}^P \mathbf{H}^T\mathbf{Y}_p\mathbf{B}\mathbf{T}\mathbf{D}_p(\mathbf{S}_E)) \quad (8.2.31)$$

更新 $\mathbf{\Gamma}$ 前要考虑到

$$(\mathbf{S}_E \odot (\mathbf{B}\mathbf{T}))^T(\mathbf{S}_E \odot (\mathbf{B}\mathbf{T})) = (\mathbf{T}^T\mathbf{B}^T\mathbf{B}\mathbf{T}) \oplus (\mathbf{S}_E^T\mathbf{S}_E)$$

类似地, 更新 \mathbf{B} , \mathbf{H} 和 \mathbf{S}_E 为

$$\mathbf{B} = (\sum_{p=1}^P \mathbf{Y}_p^T \mathbf{H}\mathbf{\Gamma}\mathbf{D}_p(\mathbf{S}_E)\mathbf{T}^T) \{ \mathbf{T}[(\mathbf{\Gamma}^T\mathbf{H}^T\mathbf{H}\mathbf{\Gamma}) \oplus (\mathbf{S}_E^T\mathbf{S}_E)]\mathbf{T}^T \}^{-1} \quad (8.2.32)$$

$$\mathbf{H} = (\sum_{p=1}^P \mathbf{Y}_p\mathbf{B}\mathbf{T}\mathbf{D}_p(\mathbf{S}_E)\mathbf{\Gamma}^T) \{ \mathbf{\Gamma}[(\mathbf{T}^T\mathbf{B}^T\mathbf{B}\mathbf{T}) \oplus (\mathbf{S}_E^T\mathbf{S}_E)]\mathbf{\Gamma}^T \}^{-1} \quad (8.2.33)$$

$$\text{diag}^{-1}(\mathbf{D}_p(\mathbf{S}_E)) = \{(\mathbf{T}^T\mathbf{B}^T\mathbf{B}\mathbf{T}) \oplus (\mathbf{\Gamma}^T\mathbf{H}^T\mathbf{H}\mathbf{\Gamma})\}^{-1} \text{diag}^{-1}(\mathbf{\Gamma}^T\mathbf{H}^T\mathbf{Y}_p\mathbf{B}\mathbf{T}), \quad p=1, \dots, P \quad (8.2.34)$$

其中 $\text{diag}^{-1}(\mathbf{X})$ 表示提取 \mathbf{X} 的对角线元素为一个列向量。

本节提出了多径衰落信道下 CDMA 系统 PARALIND 盲多用户检测算法, 算法步骤如下:

- ① 初始化 \mathbf{B} 、 \mathbf{H} 和 \mathbf{S}_E 。相关矩阵 $\mathbf{\Gamma}$ 和 \mathbf{T} 为已知的。
- ② 根据式 (8.2.32) 更新矩阵 \mathbf{B} 。
- ③ 根据式 (8.2.33) 更新矩阵 \mathbf{H} 。
- ④ 根据式 (8.2.34) 更新矩阵 \mathbf{S}_E 。
- ⑤ 重复步骤②~④直到收敛。
- ⑥ 选择得到的估计矩阵 \mathbf{B} 来实现盲多用户检测。

8.2.3 仿真结果

令 $\tilde{\mathbf{Y}}_p = \mathbf{H}\mathbf{\Gamma}\mathbf{D}_p(\mathbf{S}_E)\mathbf{T}^T\mathbf{B}^T + \mathbf{N}_p, p=1, 2, \dots, P$ 为接收的含噪信号, 其中 \mathbf{N}_p 为 AWGN 矩阵。

定义信噪比 SNR 为

$$\text{SNR} = 10 \log_{10} \frac{\sum_{p=1}^P \|HFD_p(S_E)T^T B^T\|_F^2}{\sum_{p=1}^P \|N_p\|_F^2} \text{dB} \quad (8.2.35)$$

利用 Monte Carlo 仿真来评估本节算法的比特误差率 (BER), Monte Carlo 仿真的次数为 1000。对比 PARALIND 算法与非盲空时最小均方误差 (ST-MMSE) 和空时匹配滤波器 (ST-MF), ST-MMSE 提供了盲算法的性能下界。与盲 PARALIND 算法相比, 非盲 ST-MMSE 假设拥有 DOA 的全部信息、SNR 和信道衰落信息, 而 PARALIND 算法不需要 DOA 信息和信道衰落信息。

基站使用天线数为 8 的圆阵, 扩频增益为 64, 在仿真中假设采用 BPSK 调制信号和多径衰落信道。

仿真 1: 仿真中采用 3 个用户, 每个用户到达基站多径数为 2, 算法性能如图 8.2.2~图 8.2.4 所示。

图 8.2.2 显示了在 $J=200$ 的大样本下的仿真结果。从图 8.2.2 可以发现, 盲 PARALIND 算法非常接近非盲 ST-MMSE, 并且 PARALIND 算法比 ST-MF 有更好的性能。图 8.2.3 和图 8.2.4 分别描述了在 $J=100$ 和 $J=50$ 时仿真结果。从图 8.2.2 和图 8.2.4 可以发现, 随着 J 的下降, 盲 PARALIND 和非盲 ST-MMSE 接收器的性能差距增加了。

仿真 2: 本仿真描述了 PARALIND 算法在不同用户数 K 情况下的算法性能。图 8.2.5 表示在 $J=100$ 和不同 K 下的仿真结果, 从图 8.2.5 可知, PARALIND 算法性能随着用户数的增加而下降。

仿真 3: 阵列误差向量为 $\mathbf{g}=[1, 0.6071-0.6953i, 1.0083+0.9059i, 0.3497-0.7167i, 0.9693+1.2916i, 0.5343+0.3883i, 0.7330+1.5894i, 0.8878-1.5133i]$ 。假设阵列响应为 α , 因此带有阵列误差的阵列响应向量为 $\text{diag}(\mathbf{g})\alpha$ 。采样数 J 为 100, 用户数为 4。图 8.2.6 显示了在阵列误差情况下盲 PARALIND 算法的性能。图 8.2.6 表明, 在阵列误差情况下, 盲 PARALIND 算法也有很好的性能, 盲 PARALIND 算法对于阵列误差有健壮性。

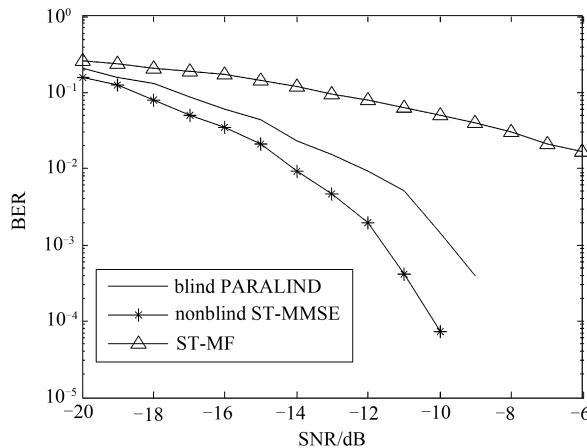
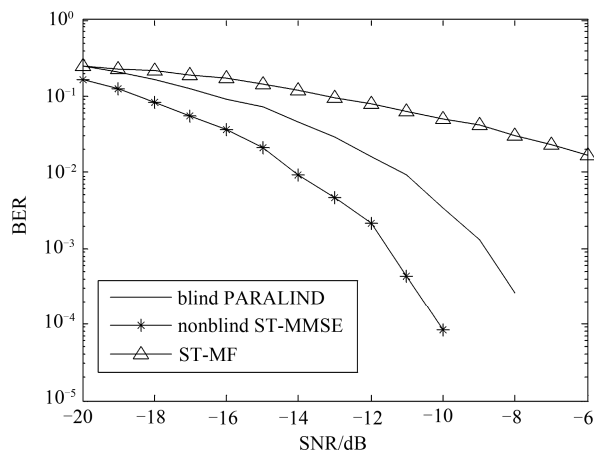
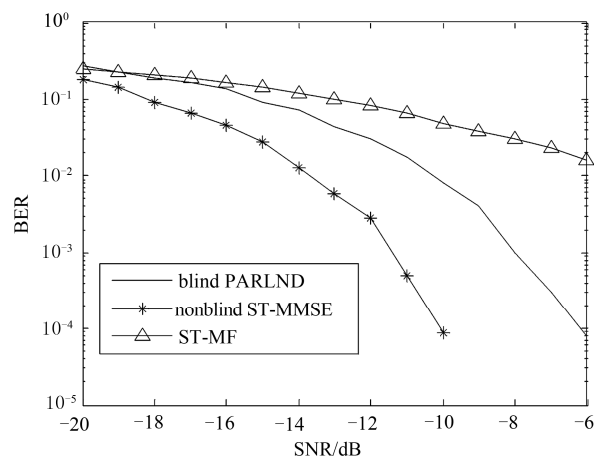
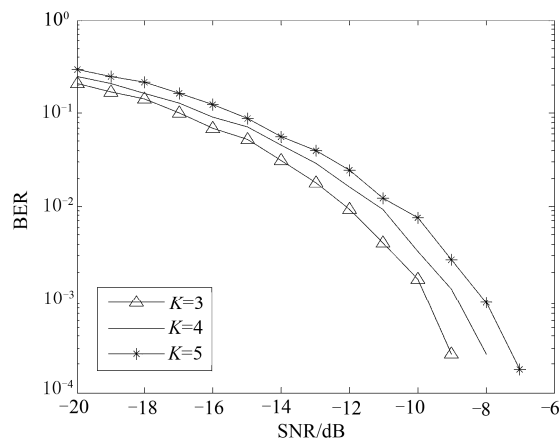


图 8.2.2 $K=4$ 和 $J=200$ 下算法性能对比

图 8.2.3 $K=4$ 和 $J=100$ 下算法性能对比图 8.2.4 $K=4$, $J=50$ 下的算法性能对比图 8.2.5 不同 K 下的算法性能对比

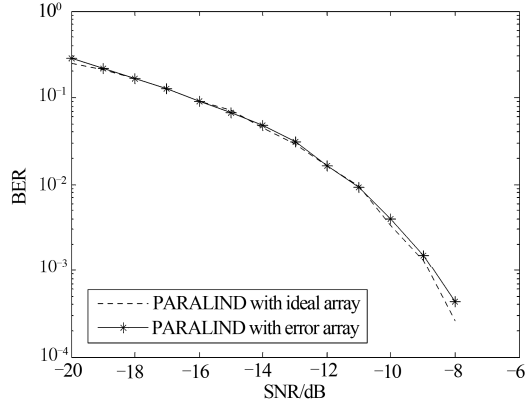


图 8.2.6 阵列误差下的算法性能

8.3 MIMO-OFDM 系统中基于 PARALIND 模型的盲信号检测

本节基于 PARALIND 模型推导一种 MIMO-OFDM 系统中的信号盲检测方法，即盲 PARALIND 算法，该算法与非盲 MMSE 算法的性能非常接近，并且在小块拍下也可以工作。此外，本节算法不需要任何信道状态信息（Channel State Information, CSI）。

8.3.1 数据模型

考虑一个 MIMO-OFDM 系统上行链路，发射和接收端分别有 I 个和 M 个天线。为了简化模型，假设接收机完全同步且已经估计并补偿了载波频偏。载波个数为 N ，采用 L 个抽样间隔的循环前缀，其中 L 的选择超过信道最大时延扩展。将第 i 个天线发射的第 k 块信号表示为 $\mathbf{s}_i(k) \in \mathbb{C}^{N \times 1}$ 。假设发射符号来自于有限符号集，对第 k 块信号进行多载波调制（采用快速傅里叶逆变换）后加上循环前缀。MIMO-OFDM 的发射机如图 8.3.1 所示，发射信号通过多径衰落信道后被包含 M 个接收天线的接收机接收。

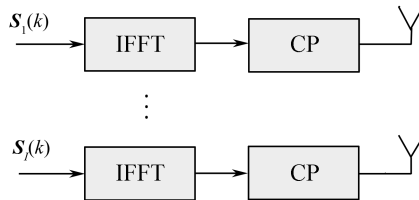


图 8.3.1 MIMO-OFDM 系统的发射机

定义 $h_{m,i}(n) = \sum_{l=0}^{L_m-1} a_{m,i}(l) e^{-j2\pi nl/N}$ 为第 n 个子载波在第 i 个发射天线和第 m 个接收天线的信道频率响应, 其中 $a_{m,i}(l)$ 表示对应的第 l 个路径的信道衰落, L_m 为多径数。其频域信道向量为

$$\mathbf{h}_{m,i} = [h_{m,i}(1), h_{m,i}(2), \dots, h_{m,i}(N)] \in \mathbb{C}^{1 \times N} \quad (8.3.1)$$

MIMO 系统中的频域信道矩阵可以表示为

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_{1,1} & \mathbf{h}_{1,2} & \cdots & \mathbf{h}_{1,I} \\ \mathbf{h}_{2,1} & \mathbf{h}_{2,2} & \cdots & \mathbf{h}_{2,I} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}_{M,1} & \mathbf{h}_{M,2} & \cdots & \mathbf{h}_{M,I} \end{bmatrix} \in \mathbb{C}^{M \times NI} \quad (8.3.2)$$

当去除 CP 后, 第 i 个发射天线和第 m 个接收天线之间的接收基带信号为

$$\mathbf{x}_{m,i}(k) = \mathbf{F}^H \text{diag}(\mathbf{h}_{m,i}) \mathbf{s}_i(k) \quad (8.3.3)$$

其中 \mathbf{F} 表示离散傅里叶变换 (DFT) 矩阵。考虑 I 个发射天线, 第 m 个接收天线接收到的无噪声信号为

$$\mathbf{x}_m(k) = \sum_{i=1}^I \mathbf{x}_{m,i}(k) = \sum_{i=1}^I \mathbf{F}^H \text{diag}(\mathbf{h}_{m,i}) \mathbf{s}_i(k) = \mathbf{F}^H \mathbf{\Gamma} \mathbf{D}_m(\mathbf{H}) \mathbf{s}(k) \quad (8.3.4)$$

其中向量 $\mathbf{s}(k) = [\mathbf{s}_1(k)^T, \mathbf{s}_2(k)^T, \dots, \mathbf{s}_I(k)^T]^T \in \mathbb{C}^{NI \times 1}$, 矩阵 $\mathbf{\Gamma}$ 为

$$\mathbf{\Gamma} = [\mathbf{I}_N \quad \mathbf{I}_N \quad \cdots \quad \mathbf{I}_N] \in \mathbb{R}^{N \times NI} \quad (8.3.5)$$

假设信道参数在 K 个信号块内为恒定的, 定义 $\mathbf{X}_m = [\mathbf{x}_m(1), \mathbf{x}_m(2), \dots, \mathbf{x}_m(K)]$, 表示成矩阵形式为

$$\mathbf{X}_m = \mathbf{F}^H \mathbf{\Gamma} \mathbf{D}_m(\mathbf{H}) \mathbf{S} \quad (8.3.6)$$

其中矩阵 \mathbf{S} 为

$$\mathbf{S} = [\mathbf{s}(1) \quad \mathbf{s}(2) \quad \cdots \quad \mathbf{s}(K)] \in \mathbb{C}^{NI \times K} \quad (8.3.7)$$

$\mathbf{\Gamma}$ 是一个相关矩阵。接收信号可以构建一个 PARALIND 模型, 在存在噪声的情况下, 接收信号为 $\tilde{\mathbf{X}}_m = \mathbf{F}^H \mathbf{\Gamma} \mathbf{D}_m(\mathbf{H}) \mathbf{S} + \mathbf{W}_m$, $m=1, 2, \dots, M$, 其中 \mathbf{W}_m 为加性高斯白噪声 (AWGN) 矩阵。

根据式 (8.3.6) 得到 $\mathbf{X}_m^H = \mathbf{S} \mathbf{D}_m^*(\mathbf{H}) \mathbf{\Gamma}^H \mathbf{F}$, $m=1, 2, \dots, M$, 接着构造下面的矩阵:

$$\begin{aligned} \mathbf{X} &= [\mathbf{X}_1^H \quad \mathbf{X}_2^H \quad \cdots \quad \mathbf{X}_M^H] \\ &= \mathbf{S}^H [\mathbf{D}_1^*(\mathbf{H}) \mathbf{\Gamma}^H \mathbf{F} \quad \mathbf{D}_2^*(\mathbf{H}) \mathbf{\Gamma}^H \mathbf{F} \quad \cdots \quad \mathbf{D}_M^*(\mathbf{H}) \mathbf{\Gamma}^H \mathbf{F}] \\ &= \mathbf{S}^H [\mathbf{H} \odot (\mathbf{F}^H \mathbf{\Gamma})]^H \end{aligned} \quad (8.3.8)$$

8.3.2 基于 PARALIND 的盲符号检测算法

1. PARALIND 分解

PARALIND 模型的代价函数为

$$\min \|\tilde{\mathbf{X}} - \mathbf{S}^H [\mathbf{H} \odot (\mathbf{F}^H \mathbf{\Gamma})]^H\|_F^2 \quad (8.3.9)$$

其中 $\tilde{\mathbf{X}}$ 为接收含噪信号。根据式 (8.3.8) 得到

$$\begin{bmatrix} X_1^H & X_2^H & \cdots & X_M^H \end{bmatrix} [H \odot (F^H \Gamma)] = S^H [H \odot (F^H \Gamma)]^H [H \odot (F^H \Gamma)] \quad (8.3.10)$$

式 (8.3.10) 的左边可以表示为

$$\begin{aligned} \begin{bmatrix} X_1^H & X_2^H & \cdots & X_M^H \end{bmatrix} [H \odot (F^H \Gamma)] &= \begin{bmatrix} X_1^H & X_2^H & \cdots & X_M^H \end{bmatrix} \begin{bmatrix} F^H \Gamma D_1(H) \\ F^H \Gamma D_2(H) \\ \vdots \\ F^H \Gamma D_M(H) \end{bmatrix} \\ &= \sum_{m=1}^M X_m^H F^H \Gamma D_m(H) \end{aligned} \quad (8.3.11)$$

根据式 (8.3.10) 和式 (8.3.11), 矩阵 S 的更新如下:

$$\hat{S}^H = \left(\sum_{m=1}^M X_m^H F^H \Gamma D_m(H) \right) ([H \odot (F^H \Gamma)]^H [H \odot (F^H \Gamma)])^{-1} \quad (8.3.12)$$

考虑到 $[H \odot (F^H \Gamma)]^H [H \odot (F^H \Gamma)] = (\Gamma^H F F^H \Gamma) \oplus (H^H H)$, 矩阵 S 的更新也可以表示为

$$\hat{S}^H = \left(\sum_{m=1}^M \tilde{X}_m^H F^H \Gamma D_m(H) \right) ((\Gamma^H F F^H \Gamma) \oplus (H^H H))^{-1} \quad (8.3.13)$$

其中 \tilde{X}_m 是含噪切片。

根据式 (8.3.6), 最小二乘法满足

$$\min \|\tilde{X}_m - F^H \Gamma D_m(H) S\|_F^2, \quad m=1, 2, \dots, M$$

相似地, H 矩阵的更新为

$$\text{diag}^{-1}(D_m(\hat{H})) = [(S S^H) \oplus (\Gamma^H F F^H \Gamma)]^{-1} \text{diag}^{-1}(\Gamma^H F X_m S^H), \quad m=1, 2, \dots, M \quad (8.3.14)$$

定义 $E = \tilde{X} - \hat{S}^H [\hat{H} \odot (F^H \Gamma)]^H$, 其中 \hat{H} 和 \hat{S} 分别表示 H 和 S 的估计。PARALIND 模型的平方冗余和 (SSR) 定义为 $\text{SSR} = \sum_{i=1}^N \sum_{j=1}^{KM} |e_{ij}|^2$, 其中 e_{ij} 是 E 的第 (i, j) 个元素。

至此, 已经推导出 MIMO-OFDM 系统中基于 PARALIND 的盲符号检测方法, 主要步骤如下。

- ① 初始化矩阵 H 和 S 。注意矩阵 F 和 Γ 是已知的。
- ② 根据式 (8.3.13) 更新矩阵 S 。
- ③ 根据式 (8.3.14) 更新矩阵 H 。
- ④ 重复步骤②和③直到收敛。

根据式 (8.3.13) 和式 (8.3.14), 矩阵 S 和 H 分别由最小二乘法更新, 矩阵不断更新直到收敛为止, 而且已知矩阵 F 和 Γ 会加快算法的收敛。

2. PARALIND 分解的唯一性

式 (8.3.5) 中的两个切片 X_i 和 X_j ($i \neq j$) 可以表示为

$$X_i = F_E S_E, \quad X_j = F_E D S_E \quad (8.3.15)$$

其中 $F_E = F^H \Gamma$, $S_E = D_i(H) S$, $D = D_i(H)^{-1} D_j(H)$ 。接着构造矩阵

$$\begin{bmatrix} X_i \\ X_j \end{bmatrix} = \begin{bmatrix} F_E \\ F_E D \end{bmatrix} S_E \quad (8.3.16)$$

S_E 行满秩保证了 $\text{span}(U) = \text{span}\left(\begin{bmatrix} F_E \\ F_E D \end{bmatrix}\right)$, 其中 U 是 $\begin{bmatrix} X_i \\ X_j \end{bmatrix}$ 的前 NI 个左奇异向量。 U 可以表示为

$$U = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} F_E T \\ F_E D T \end{bmatrix} \quad (8.3.17)$$

其中 T 是一个非奇异矩阵。构造以下自相关矩阵和互相关矩阵:

$$R_1 = U_1^H U_1 = T^H F_E^H F_E T = G T \quad (8.3.18)$$

$$R_2 = U_1^H U_2 = T^H F_E^H F_E D T = G D T \quad (8.3.19)$$

其中 $G = T^H F_E^H F_E$, R_1 和 R_2 满秩。根据式 (8.3.18) 和式 (8.3.19) 得到

$$(R_2 R_1^{-1})^T (G^{-1})^T = (G^{-1})^T D \quad (8.3.20)$$

D 和 $(G^{-1})^T$ 分别为矩阵 $(R_2 R_1^{-1})^T$ 的特征值和对应的特征向量, 且 D 唯一, 因此不考虑列模糊和尺度模糊的情况下 $(G^{-1})^T$ 可以被恢复。得到 T 、 F_E 和 S_E 的步骤如下:

- ① 根据 $R_1 = G T$ 得到 $T = G^{-1} R_1$;
- ② 根据 $U_1 = F_E T$ 得到 $F_E = U_1 T^{-1}$;
- ③ 根据 $X_i = F_E S_E$ 得到 $S_E = (F_E)^+ X_i$ 。

显然, 尺度模糊和列模糊在盲分离问题中是存在的。列模糊可以根据已知矩阵 F 和 T 来解决, 而尺度模糊可以通过加入额外信息, 或通过差分编码/解码以及自动增益控制来解决。

3. 复杂度分析

根据式 (8.3.8), 非盲 MMSE 解为

$$\hat{S}_{\text{mmse}}^H = \tilde{X} A (A^H A + I_{NI} \frac{1}{\text{SNR}})^{-1} \quad (8.3.21)$$

其中 $A = [H \odot (F^H T)]$, I_{NI} 是一个 $NI \times NI$ 的单位矩阵, 信噪比 (SNR) 定义在式 (8.3.22) 中。

与非盲 MMSE 方法相比, 本节算法有更高的计算复杂度。MMSE 方法的复杂度为 $O(N^2 I K M + N^3 \hat{P}^2 M + N^3 \hat{P}^3 + N^2 \hat{P}^2 K)$ 。本节的算法中, 每次迭代的复杂度为 $O(2N^2 \hat{P}^2 M + N^3 I + N^3 \hat{P}^2 + N^3 \hat{P}^3 + M N^2 K + 2M K N^2 I + M K N I + 2M N^2 \hat{P}^2 K + M N^3 I + M N^3 \hat{P}^2 + M N^3 \hat{P}^3)$, 而整个 PARALIND 分解过程大约需要 20 次迭代 (在图 8.3.2 中有体现)。

8.3.3 仿真结果

令 $\tilde{X}_m = F^H \Gamma D_m(H) S^T + W_m$, $m = 1, 2, \dots, M$ 为接收含噪数据, 其中 W_m 是 AWGN 矩阵。定义信噪比为

$$\text{SNR} = 10 \log_{10} \frac{\sum_{m=1}^M \|F^H \Gamma D_m(H) S^T\|_F^2}{\sum_{m=1}^M \|W_m\|_F^2} \text{dB} \quad (8.3.22)$$

采用 Monte Carlo 仿真评估 PARALIND 算法的误码率 (BER) 性能, 仿真次数为 2000。设 K 、 M 、 I 和 N 分别为数据块数、接收天线数、发射天线数以及子载波个数。

在第 i 个发射天线和第 m 个接收天线之间的信道特征描述为 $a_{m,i}(t) = \sum_{l=0}^{L_m-1} \rho_{m,i,l} \delta(t - \tau_{m,i,l})$, 其中 $\rho_{m,i,l}$ 和 $\tau_{m,i,l}$ 分别是第 l 个路径的复增益和时延。信道建模为独立瑞利衰落信道。这里采用二进制相移键控 (BPSK) 来调制信号。

首先, 验证本节算法的收敛性能。定义 $DSSR = SSR_i - SSR_0$, 其中 SSR_i 是第 i 次迭代的 SSR, SSR_0 为收敛条件下的 SSR。随机初始化 A_0 和 H 。图 8.3.2 给出了本节算法的收敛性能, 其中 $I=2$ 、 $M=3$ 、 $N=32$ 、 $K=100$ 和 $SNR=10\text{dB}$ 。从图 8.3.2 可以看出本节算法具有很好的收敛性。

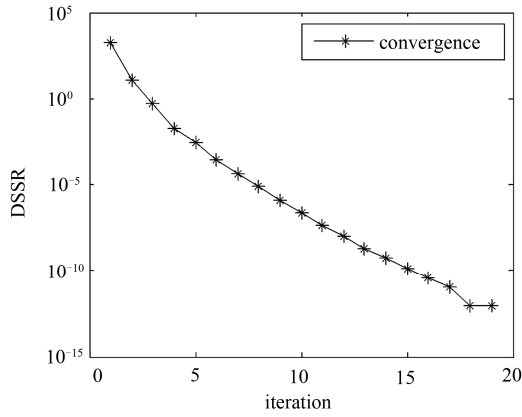


图 8.3.2 算法的收敛性能

与盲 PARALIND 接收机相比, 非盲 MMSE 方法必须假设已知准确的信道状态信息以及信噪比, 而 MMSE 方法提供了一个盲估计算法的性能下界。PARALIND 算法与非盲 MMSE 方法的性能比较如图 8.3.3 所示, 其中 $K=100$ 、 $N=32$, 图中考虑了不同的 I 和 M 。从图 8.3.3 可以看出本节算法性能非常接近非盲 MMSE 方法。

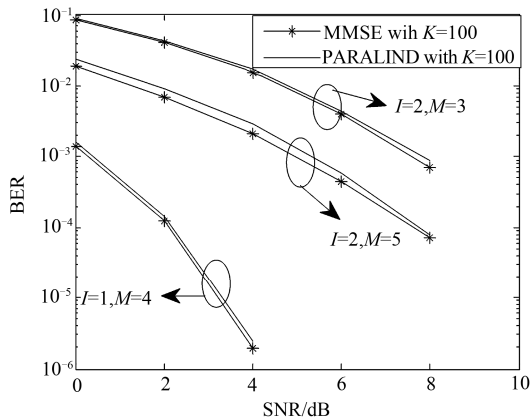


图 8.3.3 不同参数的性能比较

图 8.3.4 给出了本节算法在不同 K 下的 BER 性能比较, 其中 $I=2$ 、 $M=3$ 和 $N=32$ 。从图 8.3.4 可以看出, MMSE 方法在不同 K 下的 BER 性能非常接近, 且本节的盲 PARALIND 算法在 $K \geq 40$ 时与非盲 MMSE 方法有相近的 BER。当 $K=100$ 时, 盲 PARALIND 算法与非盲 MMSE 算法性能的差距大约为 0.5dB。从图 8.3.4 中可以看出, 盲 PARALIND 算法与非盲 MMSE 算法之间的性能差距随着 K 的增加而减小。当 K 增加时, 本节算法有更好的抑制噪声的能力。由于图 8.3.4 中同时给出了小样本 $K=10$ 下的估计性能, 可以看出本节的 PARALIND 算法在小样本下同样可以有效工作。

图 8.3.5 是本节提出的 PARALIND 算法在不同 M 下的 BER, 其中 $I=2$, $K=100$, $N=32$ 。从图 8.3.5 可以看出 PARALIND 算法的估计性能随着接收天线数增加而变好, 这是因为多天线的接收分集增益可以提升 BER 的性能。

图 8.3.6 给出了本节算法在不同的 I 下的参数估计性能, 其中 $M=4$, K 与 N 相等。从图 8.3.6 可以看出本节算法的 BER 性能随着发射天线的增加而降低, 由于发射天线增加使得天线间的互相干扰增加, 从而导致算法的 BER 性能降低。

图 8.3.7 是本节算法在不同 N 下的 BER, 其中 $I=2$, $M=3$, $K=100$ 。从图 8.3.7 可以发现, 在不同的 N 下本节算法的 BER 很接近, 在高信噪比下, 随着 N 的增大性能有一定程度的提升。

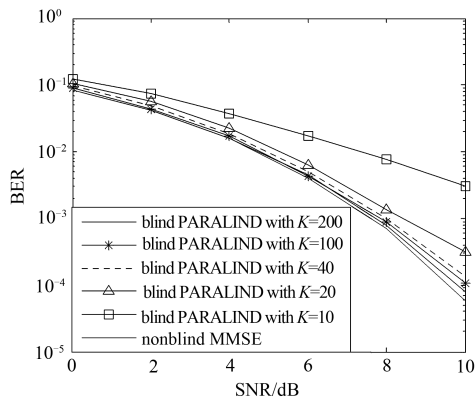


图 8.3.4 不同 K 下的性能比较

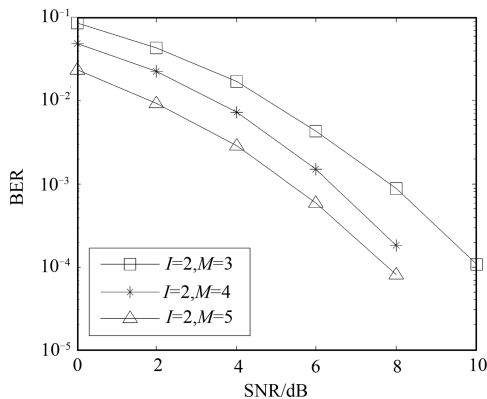


图 8.3.5 不同 M 下的 BER 比较

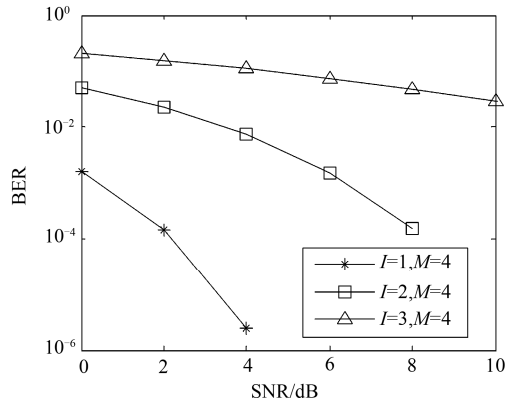


图 8.3.6 不同 I 下的 BER 性能比较

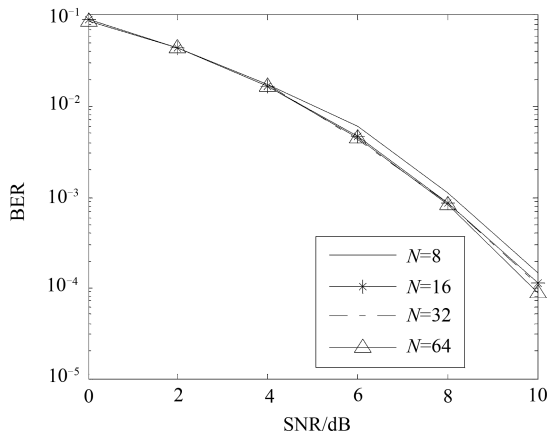


图 8.3.7 不同 N 下的 BER 性能比较

8.4 声矢量传感器阵列的基于 PARALIND 分解相干二维 DOA 估计算法

本节将声矢量传感器阵列的参数估计问题与最初应用于生物学和化学的平行线性相关剖面模型 (PARALIND) 结合在一起, 利用 PARALIND 分解方法, 针对未知位置的任意间隔的声矢量传感器阵列提出一种盲相干二维波达方向 (2D-DOA) 估计算法^[4]。该算法可以实现对声矢量传感器阵列相干角度的估计, 以及二维角度的自动配对。本节算法相比传统的相干角估计算法, 如前向后向空间平滑 FBSS-ESPRIT 算法, 具有更好的估计性能。仿真结果将验证该算法的有效性。

8.4.1 数据模型

假设有 K 个窄带平面波入射到一个含有 M 个阵元的声矢量传感器阵列上, 模型同文献[5]。假设 M 个阵列任意排列在三维空间中, 第 m 个声矢量传感器的坐标为 $r_m = (x_m, y_m, z_m)$ 。假设阵列处于空间信号辐射的远场中, 信源数为 K , 其中包括 K_1 个非相干信源和 $K - K_1$ 个相干信源, 噪声为独立于信号源的加性高斯白噪声。假设第 k 个信源的来向角为 (ϕ_k, φ_k) , 其中 ϕ_k 和 φ_k 分别代表第 k 个信号的方位角和仰角。用 $\theta_k = [\phi_k, \varphi_k]^T$ 作为第 k 个信源的二维波达方向。根据文献[6], 声矢量传感器阵的输出可以表示为

$$\mathbf{x}(t) = (\mathbf{H} \odot \mathbf{A}) \mathbf{\Gamma} \mathbf{s}(t) + \mathbf{n}(t) \quad (8.4.1)$$

其中 $\mathbf{A} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \dots, \mathbf{a}(\theta_K)] \in \mathbb{C}^{M \times K}$ 是一个 $M \times K$ 阶的方向矩阵, $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_{K_1}(t)]^T$ 是包含 K_1 个非相干信源列矢量。 $\mathbf{\Gamma} \in \mathbb{C}^{K \times K_1}$ 是相关矩阵。 $\mathbf{n}(t)$ 是接收信号的加性高斯白噪声矢量。 $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_K] \in \mathbb{C}^{4 \times K}$, 其中

$$\mathbf{h}_k = \begin{bmatrix} 1 \\ \cos \phi_k \cos \varphi_k \\ \sin \phi_k \cos \varphi_k \\ \sin \varphi_k \end{bmatrix} \quad (8.4.2)$$

因此, 收集 J 个快拍的声矢量传感器的输出信号为

$$\mathbf{X} = [\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_J)] \quad (8.4.3)$$

\mathbf{X} 可表示为

$$\mathbf{X} = (\mathbf{H} \odot \mathbf{A}) \mathbf{\Gamma} \mathbf{S} + \mathbf{N} \quad (8.4.4)$$

其中 $\mathbf{S}(t) = [s(t_1), s(t_2), \dots, s(t_J)] \in \mathbb{C}^{K_1 \times J}$, $\mathbf{N}(t) = [\mathbf{n}(t_1), \mathbf{n}(t_2), \dots, \mathbf{n}(t_J)] \in \mathbb{C}^{4M \times J}$ 为噪声矩阵。

8.4.2 相干二维角度估计

1. PARALIND 分解

定义 $\tilde{\mathbf{Y}} = \mathbf{X}^T$, 根据式 (8.4.4) 的信号模型, $\tilde{\mathbf{Y}}$ 可以表示为

$$\tilde{\mathbf{Y}} = (\mathbf{\Gamma} \mathbf{S})^T (\mathbf{H} \odot \mathbf{A})^T + \mathbf{N}^T \quad (8.4.5)$$

根据文献[7], 式 (8.4.5) 中最小二乘拟合为

$$\min_{\mathbf{\Gamma}, \mathbf{S}, \mathbf{H}, \mathbf{A}} \left\| \tilde{\mathbf{Y}} - (\mathbf{\Gamma} \mathbf{S})^T (\mathbf{H} \odot \mathbf{A})^T \right\|_F \quad (8.4.6)$$

在无噪声情况下, 根据式 (8.4.5), 可得

$$\begin{aligned} \mathbf{Y} &= [\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \mathbf{Y}_4] \\ &= [(\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_1(\mathbf{H}) \mathbf{A}^T, (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_2(\mathbf{H}) \mathbf{A}^T, (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_3(\mathbf{H}) \mathbf{A}^T, (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_4(\mathbf{H}) \mathbf{A}^T] \end{aligned} \quad (8.4.7)$$

其中 $\mathbf{Y}_n = (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \mathbf{A}^T \in \mathbb{C}^{J \times M}$, $n=1, 2, 3, 4$ 。然后得到

$$\text{vec}(\mathbf{Y}_n) = \text{vec}((\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \mathbf{A}^T) = (\mathbf{A} \mathbf{D}_n(\mathbf{H}) \otimes \mathbf{S}^T) \text{vec} \mathbf{\Gamma}^T \quad (8.4.8)$$

根据文献[7], 组合这些矢量可得

$$\begin{bmatrix} \text{vec}(\mathbf{Y}_1) \\ \text{vec}(\mathbf{Y}_2) \\ \text{vec}(\mathbf{Y}_3) \\ \text{vec}(\mathbf{Y}_4) \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{H}) \otimes \mathbf{S}^T \\ \mathbf{A}\mathbf{D}_2(\mathbf{H}) \otimes \mathbf{S}^T \\ \mathbf{A}\mathbf{D}_3(\mathbf{H}) \otimes \mathbf{S}^T \\ \mathbf{A}\mathbf{D}_4(\mathbf{H}) \otimes \mathbf{S}^T \end{bmatrix} \text{vec}(\mathbf{\Gamma}^T) \quad (8.4.9)$$

式 (8.4.9) 可以简记为

$$\text{vec}(\mathbf{Y}) = [(\mathbf{H} \odot \mathbf{A}) \otimes \mathbf{S}^T] \text{vec}(\mathbf{\Gamma}^T) \quad (8.4.10)$$

根据 (8.4.10) 得到

$$\text{vec}(\mathbf{\Gamma}^T) = [(\mathbf{H} \odot \mathbf{A}) \otimes \mathbf{S}^T]^+ \text{vec}(\mathbf{Y}) \quad (8.4.11)$$

由于已经得到了 $\text{vec}(\mathbf{\Gamma}^T)$, 所以可以很容易地通过矩阵转换得到 $\mathbf{\Gamma}$ 。

根据式 (8.4.6), \mathbf{S}^T 的最小二乘为

$$\mathbf{S}^T = \mathbf{Y} \left(((\mathbf{H} \odot \mathbf{A}) \mathbf{\Gamma})^T \right)^+ \quad (8.4.12)$$

由式 (8.4.7) 可得

$$\sum_{n=1}^4 \mathbf{Y}_n^H \mathbf{Y}_n = \left(\sum_{n=1}^4 \mathbf{Y}_n^H (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \right) \mathbf{A}^T = \mathbf{A}^* \left(\sum_{n=1}^4 \mathbf{D}_n^*(\mathbf{H}) (\mathbf{\Gamma} \mathbf{S})^* (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \right) \mathbf{A}^T \quad (8.4.13)$$

\mathbf{A} 列满秩, 因此可以通过下式获得 \mathbf{A}^* :

$$\begin{aligned} \mathbf{A}^* &= \left(\sum_{n=1}^4 \mathbf{Y}_n^H (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \right) \left(\sum_{n=1}^4 \mathbf{D}_n^*(\mathbf{H}) (\mathbf{\Gamma} \mathbf{S})^* (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \right)^{-1} \\ &= \left(\sum_{n=1}^4 \mathbf{Y}_n^H (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \right) \left(\sum_{n=1}^4 ((\mathbf{\Gamma} \mathbf{S})^* (\mathbf{\Gamma} \mathbf{S})^T) \oplus (\mathbf{H}^H \mathbf{H}) \right)^{-1} \end{aligned} \quad (8.4.14)$$

类似地, 有

$$(\mathbf{\Gamma} \mathbf{S})^* \mathbf{Y}_n \mathbf{A} = (\mathbf{\Gamma} \mathbf{S})^* (\mathbf{\Gamma} \mathbf{S})^T \mathbf{D}_n(\mathbf{H}) \mathbf{A}^T \mathbf{A} \quad (8.4.15)$$

提取等式两边矩阵的对角元素可以得到

$$\text{diag}^{-1}((\mathbf{\Gamma} \mathbf{S})^* \mathbf{Y}_n \mathbf{A}) = ((\mathbf{\Gamma} \mathbf{S})^* (\mathbf{\Gamma} \mathbf{S})^T) \oplus (\mathbf{A}^T \mathbf{A}) \text{diag}^{-1}(\mathbf{D}_n(\mathbf{H})) \quad (8.4.16)$$

然后可得

$$\text{diag}^{-1}(\mathbf{D}_n(\mathbf{H})) = (((\mathbf{\Gamma} \mathbf{S})^* (\mathbf{\Gamma} \mathbf{S})^T) \oplus (\mathbf{A}^T \mathbf{A}))^{-1} \text{diag}^{-1}((\mathbf{\Gamma} \mathbf{S})^* \mathbf{Y}_n \mathbf{A}) \quad (8.4.17)$$

矩阵 \mathbf{H} 可以通过 $\text{diag}^{-1}(\mathbf{D}_n(\mathbf{H}))$, $n = 1, 2, 3, 4$ 直接得到。

根据式 (8.4.11)、式 (8.4.12)、式 (8.4.14) 和式 (8.4.17), 发现 PARALIND 算法适用于如下所述的数据模型。

根据式 (8.4.11), $\text{vec}(\hat{\mathbf{\Gamma}}^T)$ 的估计可以写成

$$\text{vec}(\hat{\mathbf{\Gamma}}^T) = [(\hat{\mathbf{H}} \odot \hat{\mathbf{A}}) \otimes \hat{\mathbf{S}}^T]^+ \text{vec}(\tilde{\mathbf{Y}}) \quad (8.4.18)$$

其中 \mathbf{Y} 是含噪信号, $\hat{\mathbf{H}}$ 、 $\hat{\mathbf{A}}$ 和 $\hat{\mathbf{S}}^T$ 分别是 \mathbf{H} 、 \mathbf{A} 和 \mathbf{S}^T 的预估值。根据式 (8.4.12), \mathbf{S}^T 的最小二乘法解可以由下式得到

$$\hat{\mathbf{S}}^T = \tilde{\mathbf{Y}} \left(\left((\hat{\mathbf{H}} \odot \hat{\mathbf{A}}) \hat{\mathbf{r}} \right)^T \right)^+ \quad (8.4.19)$$

其中 $\hat{\mathbf{H}}$ 、 $\hat{\mathbf{A}}$ 和 $\hat{\mathbf{r}}$ 分别代表 \mathbf{H} 、 \mathbf{A} 、 \mathbf{r} 的预估值。根据式 (8.4.14), \mathbf{A}^* 的估计可以表示为

$$\hat{\mathbf{A}}^* = \left(\sum_{n=1}^4 \tilde{\mathbf{Y}}_n^H (\hat{\mathbf{r}} \hat{\mathbf{S}})^T \mathbf{D}_n(\hat{\mathbf{H}}) \right) \left(\sum_{n=1}^4 \left((\hat{\mathbf{r}} \hat{\mathbf{S}})^* (\hat{\mathbf{r}} \hat{\mathbf{S}})^T \right) \oplus (\hat{\mathbf{H}}^H \hat{\mathbf{H}}) \right)^{-1} \quad (8.4.20)$$

其中 $\hat{\mathbf{H}}$ 、 $\hat{\mathbf{S}}$ 和 $\hat{\mathbf{r}}$ 分别是 \mathbf{H} 、 \mathbf{S} 和 \mathbf{r} 的预估值。根据式 (8.4.17), $\mathbf{D}_n(\mathbf{H})$ 的估计值为

$$\text{diag}^{-1}(\mathbf{D}_n(\hat{\mathbf{H}})) = \left(\left((\hat{\mathbf{r}} \hat{\mathbf{S}})^* (\hat{\mathbf{r}} \hat{\mathbf{S}})^T \right) \oplus (\hat{\mathbf{A}}^T \hat{\mathbf{A}}) \right)^{-1} \text{diag}^{-1}((\hat{\mathbf{r}} \hat{\mathbf{S}})^* \tilde{\mathbf{Y}}_n \hat{\mathbf{A}}) \quad (8.4.21)$$

其中 $\hat{\mathbf{A}}$ 、 $\hat{\mathbf{S}}$ 、 $\hat{\mathbf{r}}$ 分别是 \mathbf{A} 、 \mathbf{S} 、 \mathbf{r} 的估计值。最后, 以 $\hat{\mathbf{H}}$ 表示的 \mathbf{H} 的估计值可以通过式 $\text{diag}^{-1}(\mathbf{D}_n(\hat{\mathbf{H}}))$, $n = 1, 2, 3, 4$ 得到。

由式 (8.4.18)、式 (8.4.19)、式 (8.4.20) 和式 (8.4.21) 可知, 矩阵 \mathbf{A} 、 \mathbf{H} 、 \mathbf{r} 和 \mathbf{S} 最小二乘更新, 直到收敛。

2. PARALIND 分解的唯一性

根据文献[7], 把 PARALIND 分解的唯一性应用于声矢量传感器阵列。通过矩阵的行列变换, 式 (8.4.7) 中的信号矩阵可以变换成另一种形式:

$$\begin{aligned} \mathbf{Z} &= [\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_J] \\ &= [\mathbf{H} \mathbf{D}_1((\mathbf{r} \mathbf{S})^T) \mathbf{A}^T, \mathbf{H} \mathbf{D}_2((\mathbf{r} \mathbf{S})^T) \mathbf{A}^T, \dots, \mathbf{H} \mathbf{D}_J((\mathbf{r} \mathbf{S})^T) \mathbf{A}^T] \end{aligned} \quad (8.4.22)$$

上式中的 \mathbf{Z}_i 和 \mathbf{Z}_j ($i \neq j$) 分别表示

$$\mathbf{Z}_i = \mathbf{H} \mathbf{D}_i((\mathbf{r} \mathbf{S})^T) \mathbf{A}^T = \mathbf{H} \mathbf{S}_E, \quad \mathbf{Z}_j = \mathbf{H} \mathbf{D}_j((\mathbf{r} \mathbf{S})^T) \mathbf{A}^T = \mathbf{H} \mathbf{A} \mathbf{S}_E \quad (8.4.23)$$

其中 $\mathbf{S}_E = \mathbf{D}_i((\mathbf{r} \mathbf{S})^T) \mathbf{A}^T$, $\mathbf{A} = \mathbf{D}_j((\mathbf{r} \mathbf{S})^T) \mathbf{D}_i^{-1}((\mathbf{r} \mathbf{S})^T)$ 。由此组成下面的矩阵:

$$\begin{bmatrix} \mathbf{Z}_i \\ \mathbf{Z}_j \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H} \mathbf{A} \end{bmatrix} \mathbf{S}_E \quad (8.4.24)$$

\mathbf{H} 要求行满秩以保证 $\text{span}(\mathbf{U}) = \text{span} \left(\begin{bmatrix} \mathbf{H} \\ \mathbf{H} \mathbf{A} \end{bmatrix} \right)$, 这里 \mathbf{U} 由矩阵 $\begin{bmatrix} \mathbf{Z}_i \\ \mathbf{Z}_j \end{bmatrix}$ 的最大的 K 个奇异

矩阵组成。 \mathbf{U} 可以表示为

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H} \mathbf{A} \end{bmatrix} \mathbf{T} \quad (8.4.25)$$

\mathbf{T} 是一个非奇异矩阵, 构建自相关矩阵和互相关矩阵如下:

$$\begin{aligned} \mathbf{R}_1 &= \mathbf{U}_1^H \mathbf{U}_1 = \mathbf{T}^H \mathbf{H}^H \mathbf{H} \mathbf{T} = \mathbf{G} \mathbf{T} \\ \mathbf{R}_2 &= \mathbf{U}_1^H \mathbf{U}_2 = \mathbf{T}^H \mathbf{H}^H \mathbf{H} \mathbf{A} \mathbf{T} = \mathbf{G} \mathbf{A} \mathbf{T} \end{aligned} \quad (8.4.26)$$

其中 $\mathbf{G} = \mathbf{T}^H \mathbf{H}^H \mathbf{H}$, \mathbf{R}_1 和 \mathbf{R}_2 是满秩矩阵, 由式 (8.4.26), 得到

$$\mathbf{R}_2 \mathbf{R}_1^{-1} \mathbf{G} = \mathbf{G} \mathbf{A} \quad (8.4.27)$$

其中 \mathbf{A} 和 \mathbf{G} 由矩阵 $\mathbf{R}_2 \mathbf{R}_1^{-1}$ 的特征值和特征向量组成。 \mathbf{A} 是唯一的。 $\mathbf{T} = \mathbf{G}^{-1} \mathbf{R}_1$, $\mathbf{H} = \mathbf{U}_1 \mathbf{T}^{-1}$, $\mathbf{S}_E = \mathbf{H}^+ \mathbf{Z}_i$, $\mathbf{A}^T = \mathbf{D}_i^{-1}((\mathbf{r} \mathbf{S})^T) \mathbf{S}_E$ 。显然, 尺度模糊和列模糊是分离问题所固有的。然而, 尺度模糊可以通过归一化轻易地消除, 列模糊在角度估计中可以忽略, 不予考虑。

3. 二维角度估计

通过 PARALIND 分解, 得到了矩阵的估计值 $\hat{\mathbf{H}} = [\hat{\mathbf{h}}_1, \hat{\mathbf{h}}_2, \dots, \hat{\mathbf{h}}_K]$, 根据式 (8.4.2), $\hat{\mathbf{h}}_k$ 可以表示为

$$\hat{\mathbf{h}}_k = [1, \cos \hat{\phi}_k \cos \hat{\phi}_k, \sin \hat{\phi}_k \cos \hat{\phi}_k, \sin \hat{\phi}_k]^T \quad (8.4.28)$$

其中 $\hat{\mathbf{h}}_k$ 是 $\hat{\mathbf{H}}$ 的第 k 个列向量, 归一化后, 通过下式易得第 k 个信源的仰角估计值为

$$\hat{\phi}_k = \sin^{-1}(\hat{\mathbf{h}}_k(4)) \quad (8.4.29)$$

第 k 个信源的方位角估计值为

$$\hat{\phi}_k = \text{angle}(\hat{\mathbf{h}}_k(2) + j\hat{\mathbf{h}}_k(3)) \quad (8.4.30)$$

显然, 方位角和仰角是自动配对的。也可以通过 PARALIND 分解得到相关矩阵的估计 $\hat{\mathbf{\Gamma}}$, 因为 $\mathbf{\Gamma}$ 的元素为 0 或 1, 所以可以轻易地通过一个判决函数获得准确的估计矩阵 $\hat{\mathbf{\Gamma}}$ 。

4. 算法步骤

声矢量传感器阵列基于 PARALIND 的二维相干 DOA 估计算法的具体步骤总结如下。

步骤 1: 从接收信号得到 $\tilde{\mathbf{Y}}$, 然后初始化矩阵 \mathbf{H} 、 \mathbf{A} 、 $\mathbf{\Gamma}$ 和 \mathbf{S} 。

步骤 2: 根据式 (8.4.18)、式 (8.4.19)、式 (8.4.20) 和式 (8.4.21) 更新 \mathbf{H} 、 \mathbf{A} 、 $\mathbf{\Gamma}$ 和 \mathbf{S} 直至收敛, 最后得到相应的估计值 $\hat{\mathbf{H}}$ 、 $\hat{\mathbf{A}}$ 、 $\hat{\mathbf{\Gamma}}$ 、 $\hat{\mathbf{S}}$ 。

步骤 3: 由式 (8.4.28) ~ 式 (8.4.30) 估计二维波达方向, 然后通过判决获得精确估计值 $\hat{\mathbf{\Gamma}}$ 。

本节算法中, 每次迭代的复杂度为 $O(4MJ(2K^2K_1^2 + 2KK_1 + 2K + K_1) + 4M(2K_1^2 + K^2 + KK_1 + 2K) + K^2(5J + M + 22) + K^3K_1^3 + 2K^3 + K_1^3 + 5KK_1J)$ 。

注 8.4.1: 本节算法通过 PARALIND 分解算法得到相关矩阵 $\mathbf{\Gamma}$ 的估计值。即使对于部分相干源, 得到的相关矩阵 $\mathbf{\Gamma}$ 结果也比较好。

注 8.4.2: PARALIND 分解算法会导致估计出 $\hat{\mathbf{H}}$ 、 $\hat{\mathbf{A}}$ 和 $\hat{\mathbf{\Gamma}}$ 具有相同的列模糊, 因此仰角、方位角以及相关矩阵都是自动配对的。

注 8.4.3: 即使信源数 K 未知, 也可以进行估计。只要对式 (8.4.22) 中的 \mathbf{Z}_i 进行奇异值分解并找到最大的奇异值即可。

5. 算法优势及克拉美罗界

根据文献[8], 可以得到未知位置的声矢量传感器阵列相干角度估计的克拉美罗界:

$$\text{CRB} = \frac{\sigma^2}{2J} \left\{ \text{Re} \left[(\mathbf{D}^H \mathbf{\Pi}_G^\perp \mathbf{D}) \oplus \mathbf{P}^T \right] \right\}^{-1} \quad (8.4.31)$$

其中 σ^2 表示噪声的方差, J 表示快拍数; $\mathbf{D} = \left[\frac{\partial \mathbf{u}_1}{\partial \phi_1}, \dots, \frac{\partial \mathbf{u}_K}{\partial \phi_K}, \frac{\partial \mathbf{u}_1}{\partial \phi_1}, \dots, \frac{\partial \mathbf{u}_K}{\partial \phi_K} \right]$, $\mathbf{u}_k = \mathbf{Q}(:, k)$,

$$\mathbf{Q} = \mathbf{H} \odot \mathbf{A}, \mathbf{\Pi}_G^\perp = \begin{bmatrix} \mathbf{\Pi}_Q^\perp & 0 \\ & \ddots \\ 0 & \mathbf{\Pi}_Q^\perp \end{bmatrix}, \mathbf{\Pi}_Q^\perp = \mathbf{I}_{4M} - \mathbf{Q}(\mathbf{Q}^H \mathbf{Q})^{-1} \mathbf{Q}^H; \mathbf{P} = \begin{bmatrix} \mathbf{P}_s & \mathbf{P}_s \\ \mathbf{P}_s & \mathbf{P}_s \end{bmatrix}, \mathbf{P}_s = \sum_{j=1}^J \mathbf{b}(t_j) \mathbf{b}(t_j)^H / J$$

且 $\mathbf{b}(t_j) = \mathbf{\Gamma} \mathbf{s}(t_j)$ 。

本节算法的优势总结起来，可以表述如下：

- ① 该算法适用于相干以及非相干信源的二维波达方向估计；
- ② 该算法不需要知道接收阵列的具体特性；
- ③ 该算法可以自动实现二维角度和相关矩阵的自动配对；
- ④ 在均匀线阵情况下，该算法角度估计性能好于 FBSS-ESPRIT 算法；
- ⑤ 该算法对于相近信源具有非常好的估计效果。

8.4.3 仿真结果

在下面的仿真中，假设共有三个信源。其中信源 1 和信源 3 是相干信源，即相关矩阵为 $\mathbf{\Gamma} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}^T$ 。

假设入射信号的角度分别为 $(\phi_1, \varphi_1) = (-15^\circ, 10^\circ)$ ， $(\phi_2, \varphi_2) = (-25^\circ, 20^\circ)$ ， $(\phi_3, \varphi_3) = (-35^\circ, 30^\circ)$ 。 M 、 J 和 K 分别代表阵元个数、快拍数以及信源个数。用 1000 次 Monte Carlo 实验的估计值来评估算法的角度估计性能。定义均方根误差 (RMSE) 为

$$\text{RMSE} = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{1000} \sum_{l=1}^{1000} (\hat{\phi}_{k,l} - \phi_k)^2 + (\hat{\varphi}_{k,l} - \varphi_k)^2} \quad (8.4.32)$$

这里 ϕ_k 和 φ_k 分别代表第 k 个信源的实际方位角和仰角， $\hat{\phi}_{k,l}$ 和 $\hat{\varphi}_{k,l}$ 是第 l 次 Monte Carlo 实验中 ϕ_k 和 φ_k 的估计值。

为了展示本节算法与 FBSS-ESPRIT 算法估计性能的比较，假设图 8.4.1 和图 8.4.2 的声矢量传感器为均匀线阵。图 8.4.1 描绘了 $M=12$ 、 $J=100$ 、 $K=3$ 和 $\text{SNR}=15\text{dB}$ 时的二维角度估计性能，证明了本节算法对于使用均匀线阵作为接收阵列时的二维角度估计是有效的。图 8.4.2 展示了 $M=12$ 、 $J=100$ 、 $K=3$ 时两种估计算法的性能对比以及对应的克拉美罗界。这表明在声矢量传感器阵为均匀线阵时本节算法的估计性能要优于 FBSS-ESPRIT 算法。

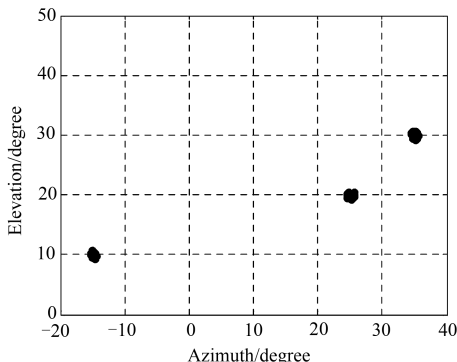


图 8.4.1 SNR=15dB 时对于均匀线阵的角度估计性能 ($M=12$, $J=100$, $K=3$)

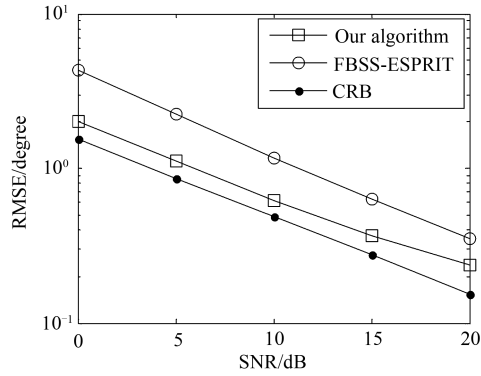


图 8.4.2 两种算法的角度估计性能比较 ($M=12$, $J=100$ 和 $K=3$)

接下来的图 8.4.3~图 8.4.7 是针对处于未知位置的任意声矢量传感器阵列的情况, 假设列满秩的接收方向矩阵是随机生成的。图 8.4.3 展示了 $M=12$ 、 $J=100$ 、 $K=3$ 和 $\text{SNR}=5$ dB 时本算法的二维角度估计性能, 图 8.4.4 描绘的是 $M=12$ 、 $J=100$ 、 $K=3$ 和 $\text{SNR}=15$ dB 时的角度估计。图 8.4.3 和图 8.4.4 说明了本算法对于任意声矢量传感器阵列配对二维角度估计的有效性。图 8.4.5 所示是 $M=12$ 、 $J=100$ 和 $K=3$ 时, 本算法在不同信噪比条件下的性能比较以及克拉美罗界。图 8.4.6 描绘了 $M=10$ 、 $J=50$ 和 $K=3$ 时, 本算法在不同信噪比条件下的性能比较以及克拉美罗界。

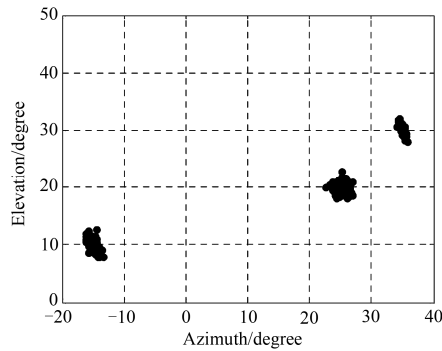


图 8.4.3 SNR=5 dB 时本算法二维角度估计 ($M=12$, $J=100$, $K=3$)

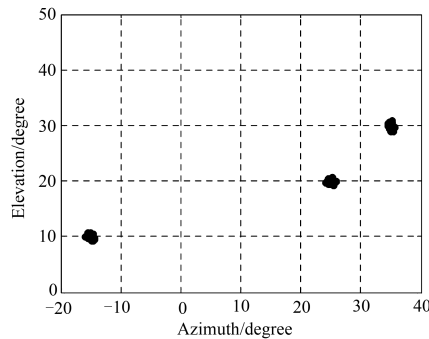


图 8.4.4 SNR=15 dB 时本算法二维角度估计 ($M=12$, $J=100$, $K=3$)

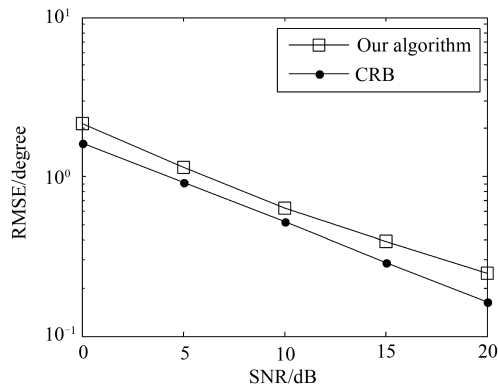
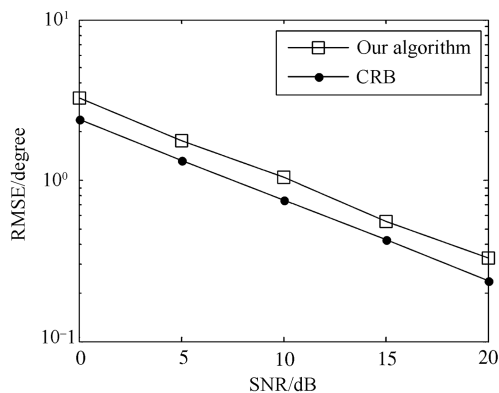
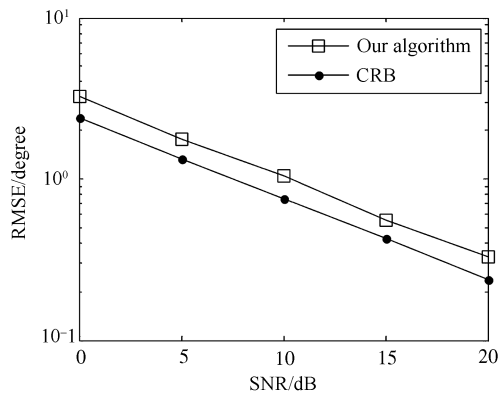
图 8.4.5 估计性能比较和克拉美罗界 ($M=12$, $J=100$, $K=3$)图 8.4.6 估计性能比较和克拉美罗界 ($M=10$, $J=50$, $K=3$)

图 8.4.7 描绘了本算法在 $M=12$ 和 $K=3$ 时, 随快拍数 J 变化时的估计性能, 证明了本算法适用于小快拍数且随着快拍数的增加估计性能越来越好。图 8.4.8 描绘了本算法在 $M=12$ 、 $J=100$ 时, 随信源数 K 变化时的估计性能比较, 表明了本算法的估计性能会随着信源数的减少而逐渐改善。

图 8.4.7 快拍数 J 不同时的估计性能比较 ($M=12$, $K=3$)

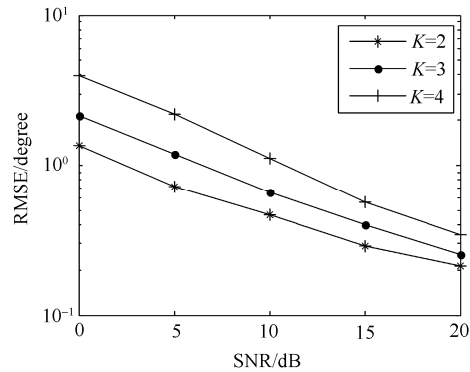


图 8.4.8 信源数 K 不同时的估计性能比较 ($M = 12, J = 100$)

假设两个相隔很近的相干信源分别位于 $(\phi_1, \varphi_1) = (0^\circ, 30^\circ)$, $(\phi_2, \varphi_2) = (2^\circ, 28^\circ)$ 。图 8.4.9 展示了当 $M = 12$ 、 $J = 100$ 、 $K = 2$ 、 $\text{SNR} = 15 \text{ dB}$ 时利用本算法对相近信源的估计性能，表明了算法对相近信源有非常好的估计性能。

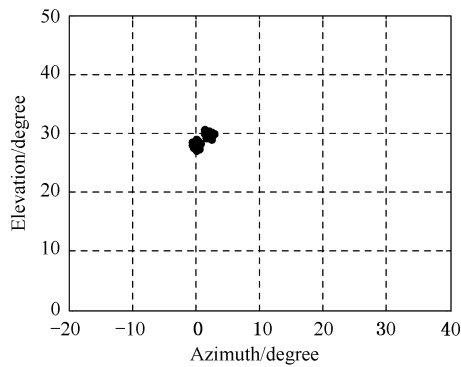


图 8.4.9 本算法对于密集信源的角度估计

本节将最初应用于生物和化学理论的 PARALIND 模型加以拓展使之应用在声矢量传感器的信号处理之中。PARALIND 模型被用于未知位置的任意声矢量传感器阵列的二维相干 DOA 中。本节已经将 PARALIND 分解算法及其唯一性应用于声矢量传感器阵列。本节算法能够获得自动配对的信源相关矩阵和二维 DOA 估计值，与用于均匀阵列的 FBSS-ESPRIT 算法相比较，本节算法具有更好的角度估计性能。而且，本节算法对于相近信源具有非常好的估计效果。

参考文献

- [1] Zhang X, Feng G, Xu D, et al. Blind PARALIND space-time multiuser detection for asynchronous CDMA system [J]. Journal of Circuits, Systems, and Computers, 2009, 18(3):

503-517.

- [2] Zhang X, Gao X, Wang Z. Blind PARALIND multiuser detection for smart antenna CDMA system over multipath fading channel [J]. Progress In Electromagnetics Research, 2009, 89: 23-38.
- [3] Xiaofei Z, Fei W, Dazhuan X. Blind signal detection algorithm for MIMO-OFDM systems over multipath channel using PARALIND model [J]. IET Communications, 2011, 5 (5) : 606-611.
- [4] Zhang X, Zhou M, Li J. A PARALIND Decomposition-Based Coherent Two-Dimensional Direction of Arrival Estimation Algorithm for Acoustic Vector-Sensor Arrays [J]. Sensors, 2013, 13 (4) : 5302-5316.
- [5] Zhang X, Li J, Chen H, et al. Trilinear decomposition-based two-dimensional DOA estimation algorithm for arbitrarily spaced acoustic vector-sensor array subjected to unknown locations [J]. Wireless Personal Communications, 2012, 67 (4) : 859-877.
- [6] Nehorai A, Paldi E. Acoustic vector-sensor array processing [J]. IEEE Transactions on Signal Processing, 1994, 42 (9) : 2481-2491.
- [7] Bro R, Harshman R A, Sidiropoulos N D, et al. Modeling multi-way data with linearly dependent loadings[J]. Journal of Chemometrics, 2009, 23 (7-8) : 324-340.
- [8] Stoica P, Nehorai A. Performance study of conditional and unconditional direction-of-arrival estimation [J]. IEEE Transactions on Acoustics, Speech and Signal Processing, 1990, 38 (10) : 1783-1795.

第9章

PARAFAC 压缩感知理论 在通信和信号处理中的应用

本章就 PARAFAC 压缩感知理论在通信和信号处理中的应用给出几个例子。

9.1 基于 PARAFAC 压缩感知模型阵列信号检测

本章将均匀面阵信号检测问题与压缩感知三线性模型联系起来^[1]。利用这个联系，提出基于压缩感知三线性分解模型的信号检测算法，利用信号稀疏性进行信号检测。

9.1.1 数据模型

一个包含 $M \times N$ 的均匀面阵，阵元间距为 d 。考虑远场信号，即接收到的信号为平面波。假设噪声与信号独立，并且假设噪声为加性同分布高斯噪声。假设有 K 个不相关信源， ϕ_k, θ_k 分别为第 k 个信源的方位角和仰角。第一个子阵列的接收信号为 $\mathbf{x}_1(t) = \mathbf{A}_x \mathbf{s}(t) + \mathbf{n}_1(t)$ ，其中 $\mathbf{A}_x = [\mathbf{a}_x(\phi_1, \theta_1), \dots, \mathbf{a}_x(\phi_K, \theta_K)]$ ， $\mathbf{a}_x(\phi_k, \theta_k) = [1, e^{-j2\pi d \sin \theta_k \cos \phi_k f_k / c}, \dots, e^{-j2\pi(M-1)d \sin \theta_k \cos \phi_k f_k / c}]^T$ ， f_k 为第 k 个信源的中心频率， c 为光速； $\mathbf{s}(t) \in \mathbb{C}^{K \times 1}$ 为信源向量， $\mathbf{n}_1(t)$ 为接收到的噪声。第 n 个子阵列接收信号表示为 $\mathbf{x}_n(t) = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{s}(t) + \mathbf{n}_n(t)$ ，其中 $\mathbf{A}_y = [\mathbf{a}_y(\phi_1, \theta_1), \dots, \mathbf{a}_y(\phi_K, \theta_K)]$ ， $\mathbf{a}_y(\phi_k, \theta_k) = [1, e^{-j2\pi d \sin \theta_k \sin \phi_k f_k / c}, \dots, e^{-j2\pi(N-1)d \sin \theta_k \sin \phi_k f_k / c}]^T$ ， $\mathbf{n}_n(t)$ 为第 n 个子阵列接收的噪声， $\mathbf{D}_i(\cdot)$ 是取矩阵的第 i 行构造的一个对角矩阵。面阵的接收信号为

$$\begin{aligned} \mathbf{x}(t) &= \begin{bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \\ \vdots \\ \mathbf{x}_N(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_x \\ \mathbf{A}_x \mathbf{D}_2(\mathbf{A}_y) \\ \vdots \\ \mathbf{A}_x \mathbf{D}_N(\mathbf{A}_y) \end{bmatrix} \mathbf{s}(t) + \begin{bmatrix} \mathbf{n}_1(t) \\ \mathbf{n}_2(t) \\ \vdots \\ \mathbf{n}_N(t) \end{bmatrix} \\ &= [\mathbf{A}_y \odot \mathbf{A}_x] \mathbf{s}(t) + \mathbf{n}(t) \end{aligned} \quad (9.1.1)$$

其中

$$\mathbf{n}(t) = \begin{bmatrix} \mathbf{n}_1(t) \\ \mathbf{n}_2(t) \\ \vdots \\ \mathbf{n}_N(t) \end{bmatrix}$$

考虑 L 个样本, 定义 $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(L)]$, 其可以表示为

$$\mathbf{X} = [\mathbf{A}_y \odot \mathbf{A}_x] \mathbf{S}^T + \mathbf{N} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_N \end{bmatrix} = \begin{bmatrix} \mathbf{A}_x \mathbf{D}_1(\mathbf{A}_y) \\ \mathbf{A}_x \mathbf{D}_2(\mathbf{A}_y) \\ \vdots \\ \mathbf{A}_x \mathbf{D}_N(\mathbf{A}_y) \end{bmatrix} \mathbf{S}^T + \begin{bmatrix} \mathbf{N}_1 \\ \mathbf{N}_2 \\ \vdots \\ \mathbf{N}_N \end{bmatrix} \quad (9.1.2)$$

其中 $\mathbf{N} = [\mathbf{n}(1), \mathbf{n}(2), \dots, \mathbf{n}(L)]$, $\mathbf{S} = [\mathbf{s}(1), \mathbf{s}(2), \dots, \mathbf{s}(L)]^T \in \mathbb{C}^{L \times K}$ 。这样, \mathbf{X}_n 可以表示为

$$\mathbf{X}_n = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{S}^T + \mathbf{N}_n, \quad n = 1, 2, \dots, N \quad (9.1.3)$$

式 (9.1.3) 的无噪声信号也可以表示为三线性模型^[2]:

$$x_{m,n,l} = \sum_{k=1}^K \mathbf{A}_x(m, k) \mathbf{A}_y(n, k) \mathbf{S}(l, k), \quad m = 1, \dots, M, \quad n = 1, \dots, N, \quad l = 1, \dots, L \quad (9.1.4)$$

其中, $\mathbf{A}_x(m, k)$ 为矩阵 \mathbf{A}_x 的第 (m, k) 个元素, 其他类似。 $\mathbf{X}_n = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{S}^T$ ($n=1, 2, \dots, N$)

为三维数据矩阵沿着一个方向的切片。式 (9.1.4) 的三线性模型对称性允许另外两种矩阵切片形式, 因此可以得到 $\mathbf{Y}_m = \mathbf{S} \mathbf{D}_m(\mathbf{A}_x) \mathbf{A}_y^T$, $m = 1, 2, \dots, M$, $\mathbf{Z}_l = \mathbf{A}_y \mathbf{D}_l(\mathbf{S}) \mathbf{A}_x^T$, $l = 1, 2, \dots, L$ 。

构造矩阵 \mathbf{Y} 和 \mathbf{Z} 如下:

$$\mathbf{Y} = [\mathbf{Y}_1^T \quad \mathbf{Y}_2^T \quad \dots \quad \mathbf{Y}_M^T]^T = [\mathbf{A}_x \odot \mathbf{S}] \mathbf{A}_y^T \quad (9.1.5)$$

$$\mathbf{Z} = [\mathbf{Z}_1^T \quad \mathbf{Z}_2^T \quad \dots \quad \mathbf{Z}_L^T]^T = [\mathbf{S} \odot \mathbf{A}_y] \mathbf{A}_x^T \quad (9.1.6)$$

9.1.2 利用三线性模型压缩感知的信号检测算法

1. 压缩

第一步, 压缩三维数组 $\bar{\mathbf{X}} \in \mathbb{C}^{M \times N \times L}$ 到一个更小的三维数组 $\bar{\mathbf{X}}' \in \mathbb{C}^{M' \times N' \times L'}$, 其中 $M' < M$, $N' < N$, $L' < L$, 如图 9.1.1 所示。定义压缩矩阵为 $\mathbf{U} \in \mathbb{C}^{M \times M'}$ 、 $\mathbf{V} \in \mathbb{C}^{N \times N'}$ 和 $\mathbf{W} \in \mathbb{C}^{L \times L'}$ 。根据文献[3], 压缩后数据可以表示为

$$\mathbf{X}' = [\mathbf{A}'_y \odot \mathbf{A}'_x] \mathbf{S}'^T \quad (9.1.7)$$

其中 $\mathbf{A}'_x = \mathbf{U}^H \mathbf{A}_x$ 、 $\mathbf{A}'_y = \mathbf{V}^H \mathbf{A}_y$ 和 $\mathbf{S}' = \mathbf{W}^H \mathbf{S}$ 。根据式 (9.1.5) 和式 (9.1.6), 得到以下矩阵

\mathbf{Y}' 和 \mathbf{Z}' :

$$\mathbf{Y}' = [\mathbf{A}'_x \odot \mathbf{S}'] \mathbf{A}'_y{}^T \quad (9.1.8)$$

$$\mathbf{Z}' = [\mathbf{S}' \odot \mathbf{A}'_y] \mathbf{A}'_x{}^T \quad (9.1.9)$$

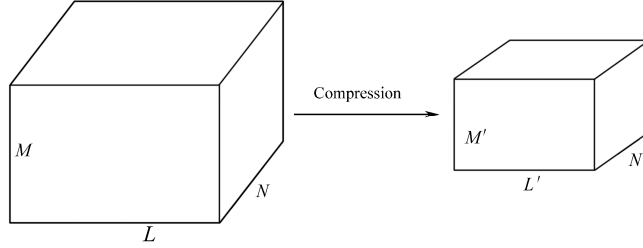


图 9.1.1 三线性模型的压缩

2. 三线性分解

TALS 算法是一个常用的数据检测方法。根据式 (9.1.7), 最小二乘拟合为

$$\min_{\mathbf{A}'_x, \mathbf{A}'_y, \mathbf{S}'} \|\tilde{\mathbf{X}}' - [\mathbf{A}'_y \odot \mathbf{A}'_x] \mathbf{S}'^T\|_F \quad (9.1.10)$$

$\tilde{\mathbf{X}}'$ 为含噪声信号。利用 LS 准则更新 \mathbf{S}' :

$$\hat{\mathbf{S}}'^T = [\hat{\mathbf{A}}'_y \odot \hat{\mathbf{A}}'_x]^+ \tilde{\mathbf{X}}' \quad (9.1.11)$$

$\hat{\mathbf{A}}'_x$ 和 $\hat{\mathbf{A}}'_y$ 分别为之前得到的 \mathbf{A}'_x , \mathbf{A}'_y 的估计。根据式 (9.1.8), LS 拟合为

$$\min_{\mathbf{A}'_x, \mathbf{A}'_y, \mathbf{S}'} \|\tilde{\mathbf{Y}}' - [\mathbf{A}'_x \odot \mathbf{S}'] \mathbf{A}'_y{}^T\|_F \quad (9.1.12)$$

$\tilde{\mathbf{Y}}'$ 为含噪声信号。利用 LS 准则更新 \mathbf{A}'_y :

$$\hat{\mathbf{A}}'_y{}^T = [\hat{\mathbf{A}}'_x \odot \hat{\mathbf{S}}']^+ \tilde{\mathbf{Y}}' \quad (9.1.13)$$

$\hat{\mathbf{A}}'_x$ 和 $\hat{\mathbf{S}}'$ 代表先前得到的 \mathbf{A}'_x 和 \mathbf{S}' 的估计值。类似地, 根据式 (9.1.9), LS 拟合为

$$\min_{\mathbf{A}'_x, \mathbf{A}'_y, \mathbf{S}'} \|\tilde{\mathbf{Z}}' - [\mathbf{S}' \odot \mathbf{A}'_y] \mathbf{A}'_x{}^T\|_F \quad (9.1.14)$$

$\tilde{\mathbf{Z}}'$ 为含噪声信号。利用 LS 准则更新 \mathbf{A}'_x :

$$\hat{\mathbf{A}}'_x{}^T = [\hat{\mathbf{S}}' \odot \hat{\mathbf{A}}'_y]^+ \tilde{\mathbf{Z}}' \quad (9.1.15)$$

$\hat{\mathbf{A}}'_y$ 和 $\hat{\mathbf{S}}'$ 代表先前得到的 \mathbf{A}'_y 和 \mathbf{S}' 的估计值。由式 (9.1.11), 式 (9.1.13) 和式 (9.1.15), \mathbf{S}' 、 \mathbf{A}'_y 和 \mathbf{A}'_x 通过最小二乘更新直到矩阵收敛。

3. 利用频率稀疏性的信号估计

经过三线性分解可以得到 \mathbf{S}' 的估计值。利用 \mathbf{s}'_k 来表示 \mathbf{S}' 的第 k 列:

$$\mathbf{s}'_k = \mathbf{W}^H \mathbf{s}_k = \mathbf{W}^H \mathbf{F}^H \mathbf{s}_k^f \quad (9.1.16)$$

其中 \mathbf{s}_k^f 为一个稀疏向量, \mathbf{s}_k 为时域形式, 矩阵 \mathbf{F} 为 DFT 矩阵。 \mathbf{s}_k^f 可以通过 l_1 范数约束得到:

$$\begin{aligned} \min \quad & \|s_k^f\|_1 \\ \text{st.} \quad & s_k' = W^H F^H s_k^f \end{aligned} \quad (9.1.17)$$

$\|s_k^f\|_1$ 为 s_k^f 的 l_1 范数形式。使用压缩感知恢复方法可以得到 s_k^f ，然后经过 IDFT 变换，信号 s_k 就可以得到了。

9.1.3 仿真结果

在以下的仿真中，采用了 $M=8$ 、 $N=6$ 的均匀面阵，并且假设三个信源从不同角度入射。中心频率分别为 $(f_1, f_2, f_3) = [1 \times 10^5 \text{ Hz}, 2 \times 10^5 \text{ Hz}, 3 \times 10^5 \text{ Hz}]$ 。采样频率为 $f_s = 2 \times 10^6 \text{ Hz}$ ，样本数为 $L=200$ 。图 9.1.2~图 9.1.4 显示了 SNR=5dB 和 SNR=15dB 下的仿真结果，可以发现信号都可以被精确地检测到，但在恢复信号中略有不同，不同幅度的原因来自于三线性模型中的尺度模糊。

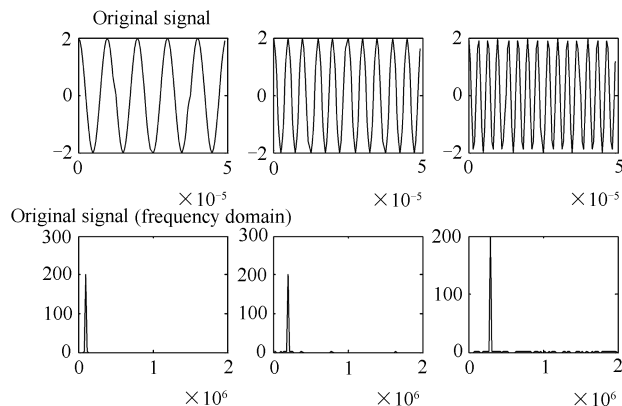


图 9.1.2 原始信号

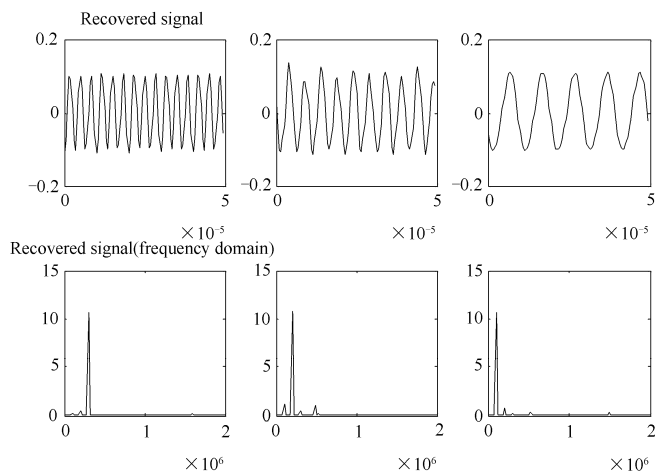


图 9.1.3 SNR=5dB 下的恢复结果

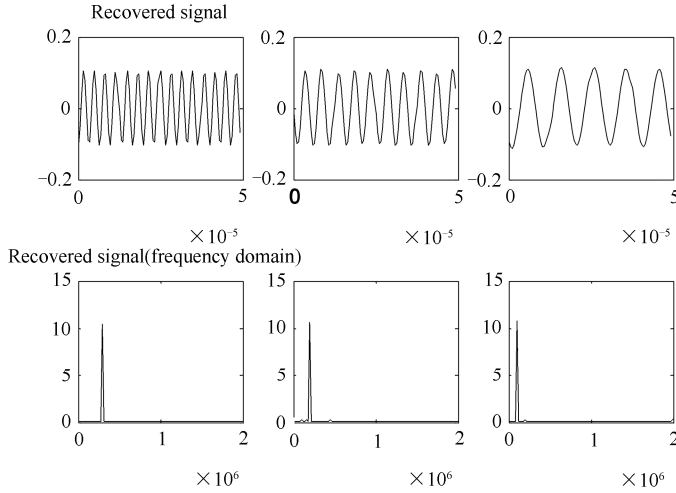


图 9.1.4 SNR=15dB 下的恢复结果

9.2 MIMO 雷达中基于压缩感知平行因子分析的联合角度与多普勒频率估计

本节提出一种压缩感知平行因子算法，可用于单基地 MIMO 雷达中联合 DOA 与多普勒频率估计^[4]。将联合角度及多普勒频率估计问题与压缩感知三线性模型相结合，减小三线性分解所需的计算量和存储空间。本节算法的角度估计性能非常接近传统的三线性分解算法，且优于 ESPRIT 算法。本节算法得到的 DOA 与多普勒频率估计值可以实现自动配对。最后本节推导单基地 MIMO 雷达中角度与多普勒频率估计的 CRB。

9.2.1 数据模型

考虑一个单基地 MIMO 雷达系统，其收发阵列均为均匀线阵，阵元间距为 d 。收发阵列分别包含 M 和 N 个天线。发射天线发射具有相同带宽 B_w 和相同中心频率 f_0 的正交波形。假设有 K 个远场目标，其间存在相干目标。定义 θ_k 为第 k 个目标的 DOA，接收阵列的接收数据矩阵可以表示为^[5,6]

$$\mathbf{x}(t) = [\mathbf{a}_r(\theta_1) \otimes \mathbf{a}_t(\theta_1), \dots, \mathbf{a}_r(\theta_K) \otimes \mathbf{a}_t(\theta_K)] \mathbf{s}(t) + \mathbf{n}(t) \quad (9.2.1)$$

其中 $\mathbf{a}_t(\theta_k) = [1, e^{-j\pi \sin \theta_k}, \dots, e^{-j(M-1)\pi \sin \theta_k}]^T$ 是发射导向矢量， $\mathbf{a}_r(\theta_k) = [1, e^{-j\pi \sin \theta_k}, \dots, e^{-j(N-1)\pi \sin \theta_k}]^T$ 为接收导向矢量， \otimes 表示 Kronecker 积。 $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_K(t)]^T \in \mathcal{R}^{K \times 1}$ ， $s_k(t) = \beta_k e^{j2\pi f_k t / f_s}$ ， β_k 、 f_k 和 f_s 分别为雷达散射截面衰落系数、第 k 个目标的多普勒频率和脉冲重复频率。 $\mathbf{n}(t) \in \mathbb{C}^{MN \times 1}$ 是独立的噪声向量，服从 0 均值方差为 $\sigma^2 \mathbf{I}$ 的复高斯分布。

定义 $\mathbf{A} = [\mathbf{a}_r(\theta_1) \otimes \mathbf{a}_t(\theta_1), \dots, \mathbf{a}_r(\theta_K) \otimes \mathbf{a}_t(\theta_K)]$, 导向矩阵 $\mathbf{A} \in \mathbb{C}^{MN \times K}$ 可以表示为

$$\mathbf{A} = \mathbf{A}_R \odot \mathbf{A}_T = \begin{bmatrix} \mathbf{A}_T \mathbf{D}_1(\mathbf{A}_R) \\ \mathbf{A}_T \mathbf{D}_2(\mathbf{A}_R) \\ \vdots \\ \mathbf{A}_T \mathbf{D}_N(\mathbf{A}_R) \end{bmatrix}$$

其中 $\mathbf{A}_T = [\mathbf{a}_t(\theta_1), \mathbf{a}_t(\theta_2), \dots, \mathbf{a}_t(\theta_K)] \in \mathcal{R}^{M \times K}$ 和 $\mathbf{A}_R = [\mathbf{a}_r(\phi_1), \mathbf{a}_r(\phi_2), \dots, \mathbf{a}_r(\phi_K)] \in \mathcal{R}^{N \times K}$ 分别是由 K 个发射和接收导向向量组成的矩阵, \odot 表示 Khatri-Rao 积, $\mathbf{D}_i(\mathbf{A}_R)$ 表示由 \mathbf{A}_R 的第 i 行元素作为对角元素组成的矩阵。

式 (9.2.1) 中的接收信号可以简化为

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \quad (9.2.2)$$

9.2.2 联合角度与多普勒频率估计

1. PARAFAC 压缩

获得 J 个快拍的接收信号数据:

$$\mathbf{X} = [\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_J)] \quad (9.2.3)$$

将 J 个抽样信号分成 P 块, 使得 $L = J - P + 1$, 则

$$\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_P \end{bmatrix} = \begin{bmatrix} [\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_L)] \\ [\mathbf{x}(t_2), \mathbf{x}(t_3), \dots, \mathbf{x}(t_{L+1})] \\ \vdots \\ [\mathbf{x}(t_P), \mathbf{x}(t_{P+2}), \dots, \mathbf{x}(t_J)] \end{bmatrix} \in \mathcal{R}^{MNP \times L} \quad (9.2.4)$$

定义

$$\boldsymbol{\Psi} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{-j2\pi\tau f_1/f_s} & e^{-j2\pi\tau f_2/f_s} & \dots & e^{-j2\pi\tau f_K/f_s} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j2\pi\tau(P-1)f_1/f_s} & e^{-j2\pi\tau(P-1)f_2/f_s} & \dots & e^{-j2\pi\tau(P-1)f_K/f_s} \end{bmatrix} \in \mathcal{R}^{P \times K} \quad (9.2.5)$$

其中 τ 是两个连续快拍之间的时延。

第 p 部分可以表示为

$$\mathbf{X}_p = \mathbf{A}\mathbf{D}_p(\boldsymbol{\Psi})\mathbf{S} + \mathbf{N}_p, \quad p = 1, \dots, P \quad (9.2.6)$$

其中 $\mathbf{S} = [\mathbf{s}(t_1), \mathbf{s}(t_2), \dots, \mathbf{s}(t_L)] \in \mathcal{R}^{K \times L}$, $\mathbf{N}_p \in \mathcal{R}^{MN \times L}$ 是第 p 部分对应的噪声矩阵, $\mathbf{D}_p(\boldsymbol{\Psi})$ 表示一个由 $\boldsymbol{\Psi}$ 的第 p 行元素作为对角元素构成的对角矩阵。

令 $H=MN$ 。忽略噪声情况下, 式 (9.2.6) 可以表示为

$$\mathbf{x}_{h,l,p} = \sum_{f=1}^K \mathbf{a}_{h,f} \mathbf{s}_{l,f} \boldsymbol{\Psi}_{p,f} \quad (h=1, 2, \dots, H, \quad l=1, 2, \dots, L, \quad p=1, 2, \dots, P) \quad (9.2.7)$$

其中, $\mathbf{a}_{h,f}$ 表示矩阵 \mathbf{A} 的第 (h, f) 个元素, 其余符号也是如此。式 (9.2.7) 满足三线性模型或 PARAFAC 分析。式 (9.2.6) 可以认为是该三线性模型在某方向的切片, 类似地, 根据式 (9.2.7) 的三维数据模型, 可以得到另外两个方向的切片为

$$\mathbf{Y}_l = \boldsymbol{\psi} \mathbf{D}_l(\mathbf{S}^T) \mathbf{A}^T (l=1, 2, \dots, L) \quad (9.2.8)$$

$$\mathbf{Z}_h = \mathbf{S}^T \mathbf{D}_h(\mathbf{A}) \boldsymbol{\psi}^T (h=1, 2, \dots, H) \quad (9.2.9)$$

式 (9.2.6)、式 (9.2.8)、式 (9.2.9) 可以详细表示为

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_P \end{bmatrix} = \begin{bmatrix} \mathbf{A} \mathbf{D}_1(\boldsymbol{\psi}) \\ \mathbf{A} \mathbf{D}_2(\boldsymbol{\psi}) \\ \vdots \\ \mathbf{A} \mathbf{D}_P(\boldsymbol{\psi}) \end{bmatrix} \mathbf{S} = [\boldsymbol{\psi} \odot \mathbf{A}] \mathbf{S} \quad (9.2.10)$$

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_L \end{bmatrix} = \begin{bmatrix} \boldsymbol{\psi} \mathbf{D}_1(\mathbf{S}^T) \\ \boldsymbol{\psi} \mathbf{D}_2(\mathbf{S}^T) \\ \vdots \\ \boldsymbol{\psi} \mathbf{D}_L(\mathbf{S}^T) \end{bmatrix} \mathbf{A}^T = [\mathbf{S}^T \odot \boldsymbol{\psi}] \mathbf{A}^T \quad (9.2.11)$$

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \\ \vdots \\ \mathbf{Z}_H \end{bmatrix} = \begin{bmatrix} \mathbf{S}^T \mathbf{D}_1(\mathbf{A}) \\ \mathbf{S}^T \mathbf{D}_2(\mathbf{A}) \\ \vdots \\ \mathbf{S}^T \mathbf{D}_H(\mathbf{A}) \end{bmatrix} \boldsymbol{\psi}^T = [\mathbf{A} \odot \mathbf{S}^T] \boldsymbol{\psi}^T \quad (9.2.12)$$

如图 9.2.1 所示, 将三维数据矩阵 $\bar{\mathbf{X}} \in \mathcal{R}^{H \times L \times P}$ 压缩成较小的三维数据矩阵 $\bar{\mathbf{X}}' \in \mathcal{R}^{H' \times L' \times P'}$, 其中 $H' < H$, $L' < L$, $P' < P$ 。令 $\mathbf{U} \in \mathcal{C}^{H \times H'}$ 、 $\mathbf{V} \in \mathcal{C}^{L \times L'}$ 和 $\mathbf{W} \in \mathcal{C}^{P \times P'}$ 表示压缩矩阵, 可以通过 TUCKER3 模型得到, 因此压缩后的矩阵 $\bar{\mathbf{X}}'$ 可以表示为^[3]

$$\bar{\mathbf{X}}' = [\boldsymbol{\psi}' \odot \mathbf{A}'] \mathbf{S}' \quad (9.2.13)$$

其中 $\mathbf{A}' = \mathbf{U}^H \mathbf{A}$, $\mathbf{S}'^T = \mathbf{V}^H \mathbf{S}^T$, $\boldsymbol{\psi}' = \mathbf{W}^H \boldsymbol{\psi}$ 。相似地可以构造压缩矩阵 $\bar{\mathbf{Y}}'$ 和 $\bar{\mathbf{Z}}'$, 根据式 (9.2.9)、式 (9.2.10),

$$\bar{\mathbf{Y}}' = [\mathbf{S}'^T \odot \boldsymbol{\psi}'] \mathbf{A}'^T \quad (9.2.14)$$

$$\bar{\mathbf{Z}}' = [\mathbf{A}' \odot \mathbf{S}'^T] \boldsymbol{\psi}'^T \quad (9.2.15)$$

三维矩阵的压缩如图 9.2.1 所示。

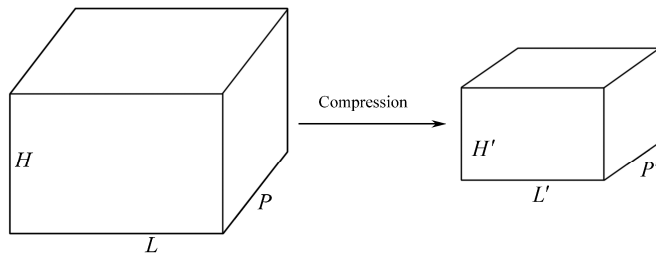


图 9.2.1 三维矩阵的压缩

2. 三线性分解

TALS 算法是三线性模型中常用的数据检测方法, 根据含噪观测数据构造低秩三线性模型进行低秩分解。三线性分解的主要 3 个步骤如下。

- ① 采用最小二乘法则更新矩阵 $\hat{\mathbf{A}}'$ 、 $\hat{\mathbf{S}}'$ 、 $\hat{\boldsymbol{\psi}}'$ 。
- ② 采用前一次估计得到的矩阵, 交替更新三个矩阵。
- ③ 不断重复上面两个步骤, 直到最小二乘代价函数收敛。

TALS 算法易于实现, 可随机初始化或采用 ESPRIT 方法初始化加快收敛。TALS 的详细过程如下。

根据式 (9.2.13), 最小二乘 LS 拟合为

$$\min_{\mathbf{A}', \boldsymbol{\psi}', \mathbf{S}'} \|\tilde{\mathbf{X}}' - [\boldsymbol{\psi}' \odot \mathbf{A}'] \mathbf{S}'\|_{\text{F}} \quad (9.2.16)$$

其中 $\tilde{\mathbf{X}}'$ 是含噪信号。采用最小二乘法更新 \mathbf{S}'

$$\hat{\mathbf{S}}'^{\text{T}} = [\hat{\boldsymbol{\psi}}' \odot \hat{\mathbf{A}}']^+ \tilde{\mathbf{X}}' \quad (9.2.17)$$

其中 $\hat{\mathbf{A}}'$ 和 $\hat{\boldsymbol{\psi}}'$ 分别为前一次得到的 \mathbf{A}' 和 $\boldsymbol{\psi}'$ 的估计值, 根据式 (9.2.14), LS 要求

$$\min_{\mathbf{A}', \boldsymbol{\psi}', \mathbf{S}'} \|\tilde{\mathbf{Y}}' - [\mathbf{S}'^{\text{T}} \odot \boldsymbol{\psi}'] \mathbf{A}'^{\text{T}}\|_{\text{F}} \quad (9.2.18)$$

其中 $\tilde{\mathbf{Y}}'$ 是含噪信号。采用最小二乘法更新 \mathbf{A}'

$$\hat{\mathbf{A}}'^{\text{T}} = [\hat{\mathbf{S}}'^{\text{T}} \odot \hat{\boldsymbol{\psi}}']^+ \tilde{\mathbf{Y}}' \quad (9.2.19)$$

其中 $\hat{\boldsymbol{\psi}}'$ 和 $\hat{\mathbf{S}}'^{\text{T}}$ 分别为前一次得到的 $\boldsymbol{\psi}'$ 和 \mathbf{S}'^{T} 的估计值。

相似地, 根据式 (9.2.15), LS 拟合为

$$\min_{\mathbf{A}', \boldsymbol{\psi}', \mathbf{S}'} \|\tilde{\mathbf{Z}}' - [\mathbf{A}' \odot \mathbf{S}'^{\text{T}}] \boldsymbol{\psi}'^{\text{T}}\|_{\text{F}} \quad (9.2.20)$$

其中 $\tilde{\mathbf{Z}}'$ 是含噪信号。采用最小二乘法更新 $\boldsymbol{\psi}'^{\text{T}}$:

$$\hat{\boldsymbol{\psi}}'^{\text{T}} = [\hat{\mathbf{A}}' \odot \hat{\mathbf{S}}'^{\text{T}}]^+ \tilde{\mathbf{Z}}' \quad (9.2.21)$$

其中 $\hat{\mathbf{A}}'$ 和 $\hat{\mathbf{S}}'^{\text{T}}$ 分别为前一次得到的 \mathbf{A}' 和 \mathbf{S}'^{T} 的估计值。

三线性分解的平方残差值 (SSR) 定义为 $\text{SSR} = \sum_{j=1}^{L'} \sum_{i=1}^{H'} |c_{ij}|$, 其中 c_{ij} 是 $\mathbf{C} = \tilde{\mathbf{X}}' - [\hat{\boldsymbol{\psi}}' \odot \hat{\mathbf{A}}'] \hat{\mathbf{S}}'$ 的第 (i, j) 个元素。 $\hat{\mathbf{S}}'$ 、 $\hat{\boldsymbol{\psi}}'$ 、 $\hat{\mathbf{A}}'$ 分别表示 \mathbf{S}' 、 $\boldsymbol{\psi}'$ 、 \mathbf{A}' 的估计值。根据式 (9.2.17)、式 (9.2.19) 和式 (9.2.21), 矩阵 \mathbf{S}' 、 $\boldsymbol{\psi}'$ 、 \mathbf{A}' 重复更新直到 SSR 收敛, 得到最终估计值 $\hat{\mathbf{S}}'$ 、 $\hat{\boldsymbol{\psi}}'$ 和 $\hat{\mathbf{A}}'$ 。

定理 9.2.1: $\mathbf{X}'_p = \mathbf{A}' \mathbf{D}_{p'}(\boldsymbol{\psi}') \mathbf{S}'$, $p' = 1, \dots, P'$, 其中 $\mathbf{A}' \in \mathcal{R}^{H' \times K}$ 、 $\mathbf{S}'^{\text{T}} \in \mathcal{R}^{L' \times K}$ 、 $\boldsymbol{\psi}' \in \mathcal{R}^{P' \times K}$ 均为满 k -秩矩阵, 如果参数满足

$$k_{\mathbf{A}'} + k_{\mathbf{S}'^{\text{T}}} + k_{\boldsymbol{\psi}'} \geq 2K + 2 \quad (9.2.22)$$

其中不考虑尺度模糊与列模糊情况下 \mathbf{A}' 、 \mathbf{S}' 、 $\boldsymbol{\psi}'$ 是唯一的。估计值 $\hat{\mathbf{A}}'$ 、 $\hat{\mathbf{S}}'$ 和 $\hat{\boldsymbol{\psi}}'$ 满足 $\hat{\mathbf{A}}' = \mathbf{A}' \boldsymbol{\Pi} \boldsymbol{\eta}_1$, $\mathbf{S}' = \mathbf{S}' \boldsymbol{\Pi} \boldsymbol{\eta}_2$, $\hat{\boldsymbol{\psi}}' = \boldsymbol{\psi}' \boldsymbol{\Pi} \boldsymbol{\eta}_3$, 其中 $\boldsymbol{\Pi}$ 是一个置换矩阵, $\boldsymbol{\eta}_1$ 、 $\boldsymbol{\eta}_2$ 、 $\boldsymbol{\eta}_3$ 为对角矩阵且满足 $\boldsymbol{\eta}_1 \boldsymbol{\eta}_2 \boldsymbol{\eta}_3 = \mathbf{I}_K$ 。

通常在三线性模型中, 有 $k_{A'} = H'$, $k_{S'^T} = K$, $k_{\psi'} = P'$, 因此式 (9.2.22) 可以等效为 $H' + K + P' \geq 2K + 2$, 可辨识条件为 $H' + P' \geq K + 2$ 。因此本节算法在 $K \leq H' + P' - 2$ 时有效, 即最大可辨识目标数为 $H' + P' - 2$ 。

3. 利用矩阵稀疏特性的联合角度与多普勒频率估计

三线性分解后, 得到 A' 、 S' 和 ψ' 的估计值。列模糊和尺度模糊在 TALS 算法中是不可避免的, 尺度模糊可以通过归一化解决, 而列模糊不影响对目标的角度估计。用 a'_k 和 ψ'_k 分别表示 A' 和 ψ' 的第 k 列, 根据压缩矩阵, 得到

$$\hat{a}'_k = U^H a_k \alpha_{ak} + n_k \quad (9.2.23a)$$

$$\hat{\psi}'_k = W^H \psi_k \alpha_{\psi k} + o_k \quad (9.2.23b)$$

其中 a_k 和 ψ_k 分别表示 A 和 ψ 的第 k 列, α_{ak} 和 $\alpha_{\psi k}$ 表示尺度系数, n_k 和 o_k 为对应的噪声。DOA 估计字典可以表示为

$$A_s = [a_s(g_1), \dots, a_s(g_b), \dots, a_s(g_B)] \in \mathbb{C}^{H \times B} \quad (9.2.24)$$

其中 $a_s(g_b) = a_{sR}(g_b) \otimes a_{sT}(g_b)$, $a_{sR}(g_b) = [1, e^{-j\pi g_b}, \dots, e^{-j(N-1)\pi g_b}]^T$, $a_{sT}(g_b) = [1, e^{-j\pi g_b}, \dots, e^{-j\pi(M-1)g_b}]^T$, $0 \leq g_1 < \dots < g_b < \dots < g_B \leq 1$, $b = 1, 2, \dots, B$, 且只能在 0 到 1 之间, 即 $\sin \theta$ 的可能取值范围。

如果 $g_{b'} = \sin \theta_k$, 得到 $a_k = A_s e_{b'}$, 其中 $e_{b'}$ 是一个 $B \times 1$ 的稀疏向量, 其第 b' 个元素为 α_{ak} , 其余为 0。式 (9.2.23a) 可以表示为

$$\hat{a}'_k = U^H A_s e_{b'} + n_k \quad (9.2.25)$$

$e_{b'}$ 可以通过 l_0 范数限制来描述:

$$\min_{e_{b'}} \|\hat{a}'_k - U^H A_s e_{b'}\|_F, \quad \text{s. t. } \|e_{b'}\|_0 = 1 \quad (9.2.26)$$

其中 $\|\cdot\|_0$ 表示 l_0 范数。式 (9.2.26) 可以写为下面的形式:

$$\min_{g(b)} \|\hat{a}'_k - U^H \alpha_{ak} a_{sb'}\|_F \quad (9.2.27)$$

其中 $a_{sb'}$ 是 A_s 的第 b' 列。

根据式 (9.2.23a), α_{ak} 可以通过 $\hat{\alpha}_{ak} = (U^H a_{sb'})^+ \hat{a}'_k$ 估计, $\sin \theta_k$ 也可以通过 l_0 范数约束来描述:

$$\sin \hat{\theta}_k = \min_{g_{b'}} \|a'_k - U^H a_{sb'} (U^H a_{sb'})^+ \hat{a}'_k\|_F, \quad b' = 1, \dots, B, \quad k = 1, \dots, K \quad (9.2.28)$$

因此角度 θ_k 可以通过下面的方法得到:

$$\hat{\theta}_k = \sin^{-1} \left(\min_{g_{b'}} \|a'_k - U^H a_{sb'} (U^H a_{sb'})^+ \hat{a}'_k\|_F \right), \quad b' = 1, \dots, B, \quad k = 1, \dots, K \quad (9.2.29)$$

频率完备字典为

$$\mathbf{F}_s = \begin{bmatrix} 1 & \cdots & 1 & \cdots & 1 \\ e^{j2\pi\tau h_1} & \cdots & e^{j2\pi\tau h_q} & \cdots & e^{j2\pi\tau h_Q} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ e^{j2\pi\tau(P-1)h_1} & \cdots & e^{j2\pi\tau(P-1)h_q} & \cdots & e^{j2\pi\tau(P-1)h_Q} \end{bmatrix} \in \mathbb{C}^{P \times Q} \quad (9.2.30)$$

其中 $0 \leq h_1 < \cdots < h_q < \cdots < h_Q \leq 1$, $q=1,2,\dots,Q$, 且只能是 0 到 1 之间, 即 f_k/f_s 的取值范围。

如果 $h_{q'} = f_k/f_s$, 有 $\boldsymbol{\psi}_k = \mathbf{F}_s \mathbf{e}_{q'}$, 其中 $\mathbf{e}_{q'}$ 是一个 $Q \times 1$ 的稀疏向量, 其第 q' 个元素为 $\alpha_{\psi k}$, 其余为 0。式 (9.2.23b) 可以表示为

$$\hat{\boldsymbol{\psi}}'_k = \mathbf{W}^H \mathbf{F}_s \mathbf{e}_{q'} + \mathbf{o}_k \quad (9.2.31)$$

$\mathbf{e}_{q'}$ 可以通过 l_0 范数约束来描述:

$$\min_{\mathbf{e}_{q'}} \|\hat{\boldsymbol{\psi}}'_k - \mathbf{W}^H \mathbf{F}_s \mathbf{e}_{q'}\|_F, \quad \text{s. t. } \|\mathbf{e}_{q'}\|_0 = 1 \quad (9.2.32)$$

重写为

$$\min_{h_{q'}} \|\hat{\boldsymbol{\psi}}'_k - \mathbf{W}^H \alpha_{\psi k} \mathbf{f}_{sq'}\|_F \quad (9.2.33)$$

其中 $\mathbf{f}_{sq'}$ 是 \mathbf{F}_s 的第 q' 列。

根据式 (9.2.23b), $\alpha_{\psi k}$ 可以通过 $\hat{\alpha}_{\psi k} = (\mathbf{W}^H \mathbf{f}_{sq'})^+ \hat{\boldsymbol{\psi}}'_k$ 来估计。 f_k/f_s 也可以通过 l_0 范数约束来描述:

$$\hat{f}_k/f_s = \min_{h_{q'}} \|\boldsymbol{\psi}'_k - \mathbf{W}^H \mathbf{f}_{sq'} (\mathbf{W}^H \mathbf{f}_{sq'})^+ \hat{\boldsymbol{\psi}}'_k\|_F, \quad q'=1,\dots,Q, \quad k=1,\dots,K \quad (9.2.34)$$

最后可以得到 f_k 的估计方法:

$$\hat{f}_k = f_s \times \min_{h_{q'}} \|\boldsymbol{\psi}'_k - \mathbf{W}^H \mathbf{f}_{sq'} (\mathbf{W}^H \mathbf{f}_{sq'})^+ \hat{\boldsymbol{\psi}}'_k\|_F, \quad q'=1,\dots,Q, \quad k=1,\dots,K \quad (9.2.35)$$

至此, 本节已经完成了基于 CS-PARAFAC 的联合角度与多普勒频率估计算法, 其主要步骤如下。

① 构造三维数据矩阵 $\bar{\mathbf{X}} \in \mathbb{C}^{H \times L \times P}$, 通过压缩矩阵 $\mathbf{U} \in \mathbb{C}^{H \times H'}$ 、 $\mathbf{V} \in \mathbb{C}^{L \times L'}$ 和 $\mathbf{W} \in \mathbb{C}^{P \times P'}$ 压缩为更小的矩阵 $\bar{\mathbf{X}}' \in \mathbb{C}^{H' \times L' \times P'}$, 根据三线性模型的对称性得到另外两个方向切片组成的矩阵 $\bar{\mathbf{Y}}'$ 和 $\bar{\mathbf{Z}}'$ 。

② 对压缩后的三维矩阵进行三线性分解, 得到 \mathbf{A}' 、 \mathbf{S}' 和 $\boldsymbol{\psi}'$ 的估计值。

③ 估计稀疏向量, 得到自动配对的 DOA 与多普勒频率估计值。

本节算法的优点:

① 相比传统的三线性分解算法, 所需的存储空间与运算复杂度更小。

② 本节算法的角度估计性能非常接近传统的三线性分解算法, 且优于 ESPRIT 算法。

③ 本节算法得到的 DOA 与多普勒频率估计值可以实现自动配对。

9.2.3 性能分析

相比传统 PARAFAC 算法, 本节算法具有更低的复杂度。本节算法的一次迭代需要计算量为 $O\{3K^3 + 2(H'P' + P'L' + HL')K^2 + 6HLP'K\}$, 而传统 PARAFAC 算法需要 $O\{3K^3 + 2(HP + PL + HL)K^2 + 6HLPK\}$, 其中 $H' < H$ 、 $L' < L$ 、 $P' < P$ (通常 \ll)。且通过压缩, 算法所需的存储空间更小, 更易于在实际应用中实现。

在这部分, 推导单基地 MIMO 雷达中角度与多普勒频率的 CRB。假设信号 $\mathbf{s}(t)$ 是确定的, 则估计参数向量可以表示为

$$\boldsymbol{\zeta} = [\theta_1, \dots, \theta_K, f_1, \dots, f_K, \text{Re}\{s(t_1)\}, \dots, \text{Re}\{s(t_J)\}, \text{Im}\{s(t_1)\}, \dots, \text{Im}\{s(t_J)\}, \sigma^2]^\top$$

收集 J 个快拍的接收信号可以表示为

$$\mathbf{Y} = [\mathbf{x}(t_1)^\top, \mathbf{x}(t_2)^\top, \dots, \mathbf{x}(t_J)^\top]^\top$$

CRB 矩阵的第 (i, j) 个元素 ζ 可以表示为

$$\text{CRB}_{ij} = \text{tr}[\boldsymbol{\Gamma}^{-1} \boldsymbol{\Gamma}'_i \boldsymbol{\Gamma}^{-1} \boldsymbol{\Gamma}'_j] + 2 \text{Re}[\boldsymbol{\mu}'^{\text{H}}_i \boldsymbol{\Gamma}^{-1} \boldsymbol{\mu}'_j]$$

其中 $\boldsymbol{\Gamma}'_i$ 、 $\boldsymbol{\Gamma}'_j$ 和 $\boldsymbol{\mu}'_i$ 、 $\boldsymbol{\mu}'_j$ 分别为 $\boldsymbol{\Gamma}$ 和 $\boldsymbol{\mu}$ 关于 $\boldsymbol{\zeta}$ 的第 i 和第 j 个元素的导数, $\text{tr}(\cdot)$ 为求矩阵迹。

由于协方差矩阵仅与 σ^2 相关联, 上式的第一部分可以忽略, \mathbf{Y} 的均值 $\boldsymbol{\mu}$ 和协方差矩阵 $\boldsymbol{\Gamma}$ 为

$$\boldsymbol{\mu} = \begin{bmatrix} \mathbf{A}\mathbf{s}(t_1) \\ \vdots \\ \mathbf{A}\mathbf{s}(t_J) \end{bmatrix} = \mathbf{G}\mathbf{S}, \quad \boldsymbol{\Gamma} = \begin{bmatrix} \sigma^2 \mathbf{I}_{MN} & & 0 \\ & \ddots & \\ 0 & & \sigma^2 \mathbf{I}_{MN} \end{bmatrix}$$

$$\text{其中 } \mathbf{G} = \begin{bmatrix} \mathbf{A} & & 0 \\ & \ddots & \\ 0 & & \mathbf{A} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{s}(t_1) \\ \vdots \\ \mathbf{s}(t_J) \end{bmatrix}。$$

可以直接得到

$$\begin{aligned} \frac{\partial \boldsymbol{\mu}_k}{\partial \theta_k} &= \begin{bmatrix} \frac{\partial \mathbf{a}_k}{\partial \theta_k} \mathbf{s}(t_1) \\ \vdots \\ \frac{\partial \mathbf{a}_k}{\partial \theta_k} \mathbf{s}(t_J) \end{bmatrix} = \begin{bmatrix} \mathbf{d}_k \mathbf{s}_k(t_1) \\ \vdots \\ \mathbf{d}_k \mathbf{s}_k(t_J) \end{bmatrix}, \quad k = 1, \dots, K \\ \frac{\partial \boldsymbol{\mu}_k}{f_k} &= \begin{bmatrix} \mathbf{A} \frac{\partial \mathbf{s}(t_1)}{f_k} \\ \vdots \\ \mathbf{A} \frac{\partial \mathbf{s}(t_J)}{f_k} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_k \mathbf{s}'_k(t_1) \\ \vdots \\ \mathbf{a}_k \mathbf{s}'_k(t_J) \end{bmatrix}, \quad k = 1, \dots, K \end{aligned}$$

其中 $\mathbf{a}_k = \mathbf{a}_r(\theta_k) \otimes \mathbf{a}_t(\theta_k)$, $\mathbf{d}_k = \frac{\partial \mathbf{a}_k}{\partial \theta_k}$, $\mathbf{s}'_k(t) = \frac{\partial \mathbf{s}_k(t)}{\partial f_k}$, $\mathbf{s}_k(t)$ 为 $\mathbf{s}(t)$ 的第 k 个元素。

定义

$$\mathbf{A}_1 = \begin{bmatrix} \mathbf{d}_1 \mathbf{s}_1(t_1) & \cdots & \mathbf{d}_K \mathbf{s}_K(t_1) \\ \vdots & \ddots & \vdots \\ \mathbf{d}_1 \mathbf{s}_1(t_J) & \cdots & \mathbf{d}_K \mathbf{s}_K(t_J) \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} \mathbf{a}_1 \mathbf{s}'_1(t_1) & \cdots & \mathbf{a}_K \mathbf{s}'_K(t_1) \\ \vdots & \ddots & \vdots \\ \mathbf{a}_1 \mathbf{s}'_1(t_J) & \cdots & \mathbf{a}_K \mathbf{s}'_K(t_J) \end{bmatrix}$$

现在有

$$\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\zeta}^T} = [\mathbf{A}_1, \mathbf{A}_2, 0]$$

则

$$2 \operatorname{Re} \left\{ \frac{\partial \boldsymbol{\mu}^H}{\partial \boldsymbol{\zeta}} \boldsymbol{\Gamma}^{-1} \frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\zeta}^T} \right\} = \begin{bmatrix} \mathbf{J} & 0 \\ 0 & 0 \end{bmatrix}$$

其中

$$\mathbf{J} \triangleq \frac{2}{\sigma^2} \operatorname{Re} \left\{ \begin{bmatrix} \mathbf{A}_1^H \\ \mathbf{A}_2^H \end{bmatrix} [\mathbf{A}_1 \quad \mathbf{A}_2] \right\}$$

\mathbf{J}^{-1} 可以写为

$$\mathbf{J}^{-1} = \frac{\sigma^2}{2} \operatorname{Re} \left(\begin{bmatrix} \mathbf{A}_1^H \mathbf{A}_1 & \mathbf{A}_2^H \mathbf{A}_1 \\ \mathbf{A}_1^H \mathbf{A}_2 & \mathbf{A}_2^H \mathbf{A}_2 \end{bmatrix} \right)^{-1}$$

进一步简化, 可以将 CRB 写为

$$\text{CRB} = \frac{\sigma^2}{2} [\operatorname{Re}(\mathbf{D}^H \mathbf{D})]^{-1}$$

其中 $\mathbf{D} = [\mathbf{S}^T \odot \mathbf{D}_1, \mathbf{D}_2 \odot \mathbf{A}]$, $\mathbf{D}_1 = \left[\frac{\partial \mathbf{a}_1}{\partial \theta_1}, \dots, \frac{\partial \mathbf{a}_K}{\partial \theta_K} \right]$, $\mathbf{D}_2 = \left[\frac{\partial \mathbf{s}_1}{\partial f_1}, \dots, \frac{\partial \mathbf{s}_K}{\partial f_K} \right]$, \mathbf{a}_k 是 \mathbf{A} 的第 k 列, \mathbf{s}_k 是 \mathbf{S} 的第 k 行。

9.2.4 仿真结果

在这部分, 采用 1000 次 Monte Carlo 仿真评估算法的角度估计性能。定义 DOA/多普勒频率的均方根误差 (RMSE) 为

$$\text{RMSE} = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{1000} \sum_{l=1}^{1000} (\hat{a}_{k,l} - a_k)^2}$$

其中, $\hat{a}_{k,l}$ 是 DOA/多普勒频率 a_k 在第 l 次仿真中的估计值。假设远场空间有 3 个非相干目

标 ($K=3$), 角度分别为 $\theta_1=10^\circ$ 、 $\theta_2=20^\circ$ 和 $\theta_3=30^\circ$ 。RCS 和多普勒频率分别为 $\beta_1=0.8$ 、 $\beta_2=0.7$ 、 $\beta_3=0.6$ 和 $f_1=1000\text{Hz}$ 、 $f_2=2000\text{Hz}$ 、 $f_3=3000\text{Hz}$ 。在所有仿真中 $f_s=10\text{kHz}$ 。

图 9.2.2 是本节的 CS-PARAFAC 算法在 $\text{SNR}=5\text{dB}$ 下的角度与多普勒频率估计结果, 其中 $M=8$ 、 $N=8$ 、 $J=100$ 和 $P=3$ 。从中可以看出本节算法在低信噪比下也可以有效估计目标角度与多普勒频率。

图 9.2.3 是本节算法在 $\text{SNR}=5\text{dB}$ 下对相近目标的估计性能, 其中 $M=N=8$ 、 $J=100$ 、 $P=4$, 三个相近目标的位置为 $\theta_1=10^\circ$ 、 $\theta_2=12^\circ$ 、 $\theta_3=14^\circ$, 多普勒频率分别为 $f_1=1000\text{Hz}$ 、 $f_2=2000\text{Hz}$ 、 $f_3=3000\text{Hz}$ 。从中可以看出本节算法在目标接近情况下仍然能够正常工作。

图 9.2.4 和图 9.2.5 是本节算法与传统三线性分解以及 ESPRIT 算法的参数估计性能比较, 其中 $M=8$ 、 $N=8$ 、 $J=100$ 、 $P=5$ 或 $P=15$ 。从图 9.2.4 和图 9.2.5 可以看出, 本节的 CS-PARAFAC 算法的参数估计性能优于 ESPRIT 算法, 且 P 较小情况下 (如 $P=5$) 角度估计性能非常接近传统的三线性分解算法, 而多普勒频率估计则与传统 PARAFAC 算法几乎相同。然后, 当 P 变大 (如 $P=15$), P' 与 P 的差距造成多普勒频率估计性能的明显下降, 但仍然接近传统 PARAFAC 算法的估计性能。总而言之, 在本节算法中, P 的值微影响多普勒频率的估计, 但几乎不影响 DOA 估计。

图 9.2.6 是本节提出的 CS-PARAFAC 算法在不同快拍数 J 下的角度与多普勒频率估计性能比较, 其中 $M=8$ 、 $N=8$ 和 $P=3$ 。从中可以看出本节算法的参数估计性能随着 J 的增加有显著的提升。

图 9.2.7 和图 9.2.8 分别为本节算法在不同 M 和 N 下的参数估计性能比较, 其中 $J=100$ 、 $P=4$ 。从中可以看出, 不管是发射阵列还是接收阵列阵元数增加, 算法的角度与多普勒频率估计性能都会提升。

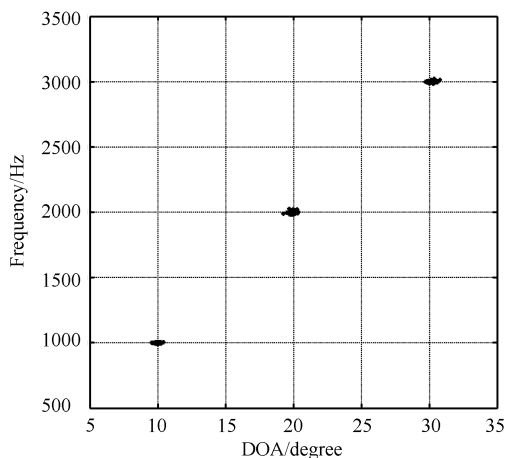


图 9.2.2 SNR=5dB 下本节算法的参数估计结果

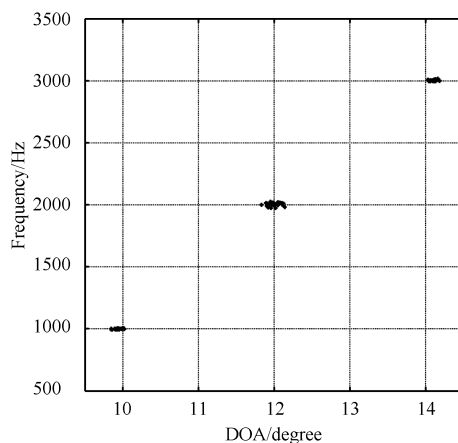
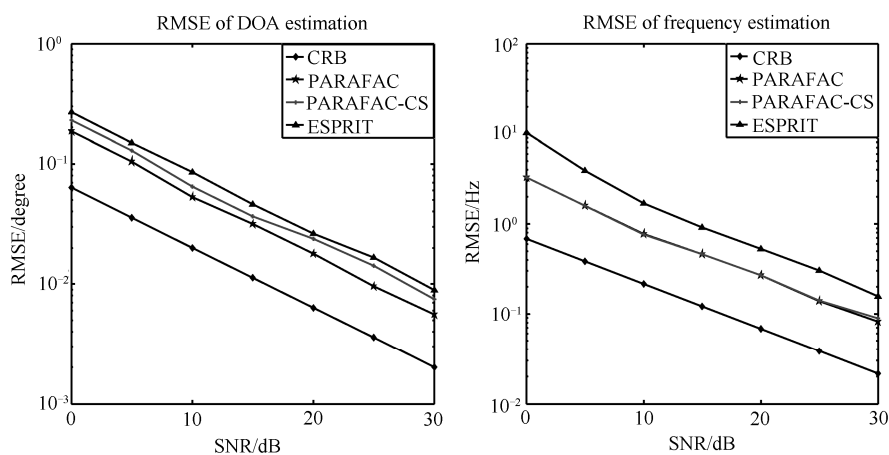
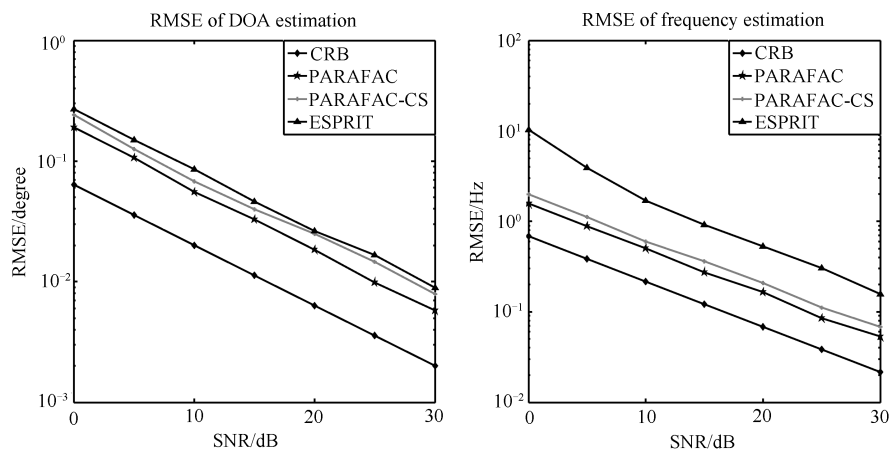


图 9.2.3 SNR=5dB 下本节算法对相近目标的参数估计结果

图 9.2.4 角度与多普勒频率估计性能比较 ($M=8$, $N=8$, $J=100$, $P=5$)图 9.2.5 角度与多普勒频率估计性能比较 ($M=8$, $N=8$, $J=100$, $P=15$)

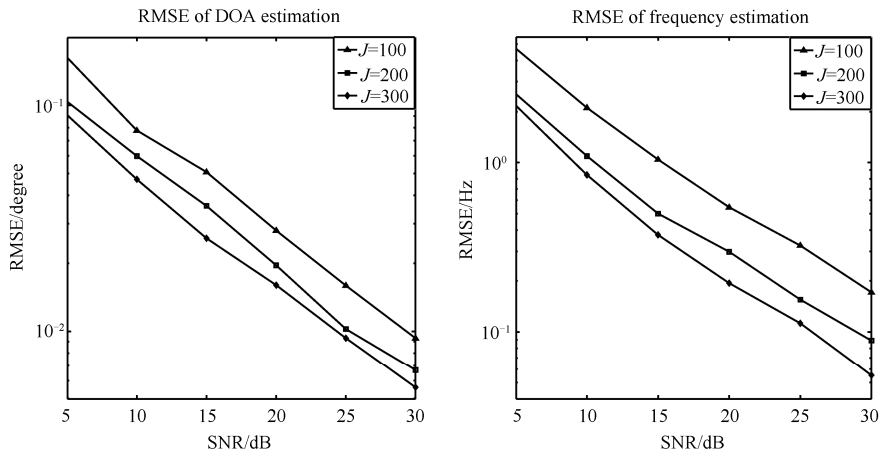


图 9.2.6 不同 J 下算法的参数估计性能 ($M=8, N=8, P=3$)

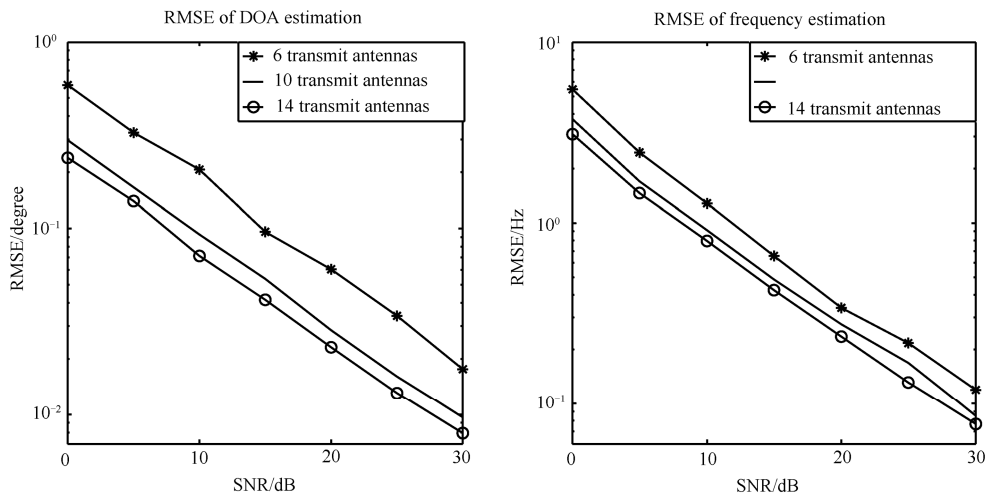


图 9.2.7 不同 M 下算法的参数估计性能 ($N=10, P=3, J=100$)

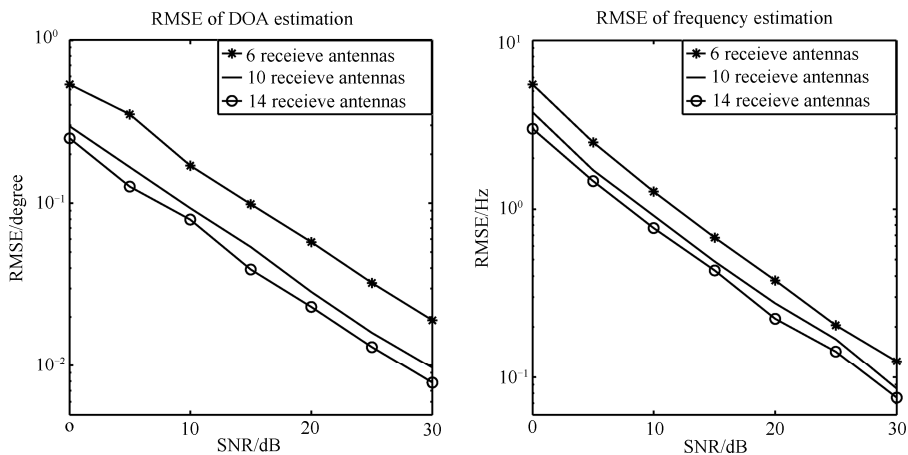


图 9.2.8 不同 N 下算法的参数估计性能 ($M=10, P=3, J=100$)

9.3 基于 PARAFAC 填充的面阵 DOA 估计

本节将讨论部分传感器无法正常工作情况下的角度估计问题。利用 OPTSPACE 算法填充部分缺失的输出矩阵，并减小噪声的影响，然后利用 PARAFAC 分解得到角度估计。

9.3.1 数据模型

一个包含 $M \times N$ 的均匀面阵如图 9.3.1 所示，阵元间距为 d 。假设有 K 个不相关信源， θ_k 、 ϕ_k 分别为第 k 个信源的仰角和方位角。第一个子阵列的接收信号为 $\mathbf{x}_1(t) = \mathbf{A}_x \mathbf{s}(t)$ ，其中 $\mathbf{A}_x = [\mathbf{a}_x(\phi_1, \theta_1), \dots, \mathbf{a}_x(\phi_K, \theta_K)]$ ， $\mathbf{a}_x(\phi_k, \theta_k) = [1, e^{-j2\pi d \sin \theta_k \cos \phi_k f_k / c}, \dots, e^{-j2\pi(M-1)d \sin \theta_k \cos \phi_k f_k / c}]^T$ ， f_k 为第 k 个信源的中心频率， c 为光速。 $\mathbf{s}(t) \in \mathbb{C}^{K \times 1}$ 为信源向量。第 n 个子阵列接收信号表示为 $\mathbf{x}_n(t) = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{s}(t)$ ，其中 $\mathbf{A}_y = [\mathbf{a}_y(\phi_1, \theta_1), \dots, \mathbf{a}_y(\phi_K, \theta_K)]$ ， $\mathbf{a}_y(\phi_k, \theta_k) = [1, e^{-j2\pi d \sin \theta_k \sin \phi_k f_k / c}, \dots, e^{-j2\pi(N-1)d \sin \theta_k \sin \phi_k f_k / c}]^T$ 。面阵的接收信号为

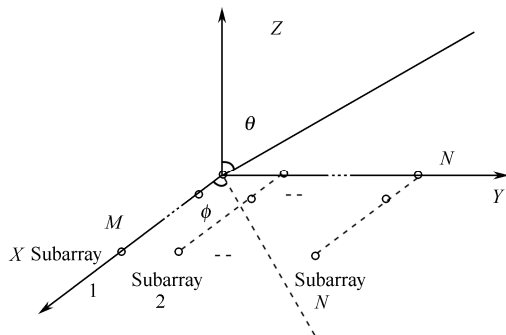


图 9.3.1 均匀面阵结构

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \\ \vdots \\ \mathbf{x}_N(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_x \\ \mathbf{A}_x \mathbf{D}_2(\mathbf{A}_y) \\ \vdots \\ \mathbf{A}_x \mathbf{D}_N(\mathbf{A}_y) \end{bmatrix} \mathbf{s}(t) = [\mathbf{A}_y \odot \mathbf{A}_x] \mathbf{s}(t) \quad (9.3.1)$$

定义 $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(L)]$ ，可表示为

$$\mathbf{X} = [\mathbf{A}_y \odot \mathbf{A}_x] \mathbf{S}^T = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_N \end{bmatrix} = \begin{bmatrix} \mathbf{A}_x \mathbf{D}_1(\mathbf{A}_y) \\ \mathbf{A}_x \mathbf{D}_2(\mathbf{A}_y) \\ \vdots \\ \mathbf{A}_x \mathbf{D}_N(\mathbf{A}_y) \end{bmatrix} \mathbf{S}^T \quad (9.3.2)$$

其中 $\mathbf{S} = [\mathbf{s}(1), \mathbf{s}(2), \dots, \mathbf{s}(L)]^T \in \mathbb{C}^{L \times K}$ 。这样， \mathbf{X}_n 可以表示为

$$\mathbf{X}_n = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{S}^T, \quad n = 1, 2, \dots, N \quad (9.3.3)$$

对于有噪声模型，信号模型变为 $\tilde{\mathbf{X}}_n = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{S}^T + \mathbf{V}_n$, $n=1, 2, \dots, N$ ，其中 \mathbf{V}_n 是接收噪声的第 n 个切片。式 (9.3.3) 也可以表示为三线性模型^[2]：

$$x_{m,n,l} = \sum_{k=1}^K \mathbf{A}_x(m, k) \mathbf{A}_y(n, k) \mathbf{S}(l, k), \quad m = 1, \dots, M, \quad n = 1, \dots, N, \quad l = 1, \dots, L \quad (9.3.4)$$

其中 $\mathbf{A}_x(m, k)$ 为矩阵 \mathbf{A}_x 的第 (m, k) 个元素，其他同理。 $\mathbf{X}_n = \mathbf{A}_x \mathbf{D}_n(\mathbf{A}_y) \mathbf{S}^T$ ($n=1, 2, \dots, N$) 为三维数据矩阵沿着一个方向的切片。根据公式 (9.3.4) 的三线性模型的对称性，可得另外两个方向切片： $\mathbf{Y}_m = \mathbf{S} \mathbf{D}_m(\mathbf{A}_x) \mathbf{A}_y^T$, $m = 1, 2, \dots, M$ ， $\mathbf{Z}_l = \mathbf{A}_y \mathbf{D}_l(\mathbf{S}) \mathbf{A}_x^T$, $l = 1, 2, \dots, L$ 。构造矩阵 \mathbf{Y} 和 \mathbf{Z} 如下：

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_M \end{bmatrix} = [\mathbf{A}_x \odot \mathbf{S}] \mathbf{A}_y^T \quad (9.3.5)$$

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \\ \vdots \\ \mathbf{Z}_L \end{bmatrix} = [\mathbf{S} \odot \mathbf{A}_y] \mathbf{A}_x^T \quad (9.3.6)$$

由于信源数 K 通常比传感器的数目 MN 和样本数 L 少，因此 $\mathbf{X} = [\mathbf{A}_y \odot \mathbf{A}_x] \mathbf{S}^T$ 的秩的大小 K 是很小的。考虑部分传感器出现随机故障时，输出 \mathbf{X} 会缺失一些值。为了简化，定义一个子集 Ω 包含 \mathbf{X} 中的非 0 元素的索引号，因此部分故障的传感器组的输出为

$$\mathbf{X}'(i, j) = \begin{cases} \mathbf{X}(i, j), & (i, j) \in \Omega \\ 0, & \text{其他} \end{cases} \quad (9.3.7)$$

9.3.2 利用 PARAFAC 填充的 DOA 估计

1. PARAFAC 填充

PARAFAC 填充法在噪声环境下可以表示为下式所示的拟合问题：

$$\begin{aligned} \min_{\mathbf{Y}} \quad & \text{rank}(\mathbf{X}) \\ \text{st.} \quad & \|\mathbf{P}_{\Omega}(\mathbf{X}') - \mathbf{P}_{\Omega}(\mathbf{X})\|_{\text{F}} < \varepsilon \end{aligned} \quad (9.3.8)$$

其中 $\mathbf{X} = [\mathbf{A}_y \odot \mathbf{A}_x] \mathbf{S}^T$ ， ε 为误差约束。 $\text{rank}(\mathbf{X})$ 表示得到矩阵 \mathbf{X} 的秩。 $\mathbf{P}_{\Omega}(\cdot)$ 表示矩阵的对应 Ω 的元素。然而这是一个非确定性 NP 难题，很多已知的算法复杂度在理论和实际中都呈双指数增长。式 (9.3.8) 中的问题可以转化为

$$\begin{aligned} \text{minimize} \quad & \|\mathbf{W}\|_* \\ \text{st.} \quad & \|\mathbf{P}_{\Omega}(\mathbf{X}') - \mathbf{P}_{\Omega}(\mathbf{W})\|_{\text{F}} < \varepsilon \end{aligned} \quad (9.3.9)$$

$\|W\|_*$ 为 W 的核范数（奇异值的和）， $\|\cdot\|_F$ 为 Frobenius 范数。

$$(P_\Omega(X'))_{i,j} = \begin{cases} X'(i,j), & (i,j) \in \Omega \\ 0, & \text{其他} \end{cases} \quad (9.3.10)$$

OPTSPACE 方法^[7]将问题转化为如下的最小值问题：

$$F(U, V) = \min_{A \in \mathbb{R}^{L \times K}} \frac{1}{2} \|P_\Omega(X' - U A V^T)\|_F^2 \quad (9.3.11)$$

其中， U 、 V 为酉矩阵， A 为对角矩阵。OPTSPACE 算法步骤如下。

① 调整。当行/列中的非 0 元素数大于均值 2 倍的时候，则称其为过表示的行/列。调整就是设置这些行和列为 0，否则奇异向量的信息会集中在这些行和列上，而不能给出 X' 中缺失元素的有用信息。假设 \tilde{X}' 为调整后的矩阵。

② 大致估算秩。当 W 的秩（信源数目）未知时，可以首先通过 \tilde{X}' 的特征值分解估算。

③ 初始化设置。 \tilde{X}' 能被分解成 $\tilde{X}' = \sum_{i=1}^{\min(MN, L)} u_i \sigma_i v_i^T$ 时，初始化设置为 $U_0 = \sqrt{MN}[u_1, \dots, u_K]$ ， $V_0 = \sqrt{L}[v_1, \dots, v_K]$ 。

④ 通过梯度下降算法最小化 $F(U, V)$ 。梯度下降算法开始于步骤③中的初始化设置，经过 n 次迭代，最优化的 A_{n+1} 可以通过 (U_n, V_n) 和 \tilde{X}' 以及最小二乘拟合得到。达到边界条件时终止。

最后，得到了 U 、 A 和 V ，估计结果为

$$\tilde{X} = U A V^T \quad (9.3.12)$$

2. PARAFAC 分解

三线性交替最小二乘（TALS）算法是一个常用的数据检测方法。根据式（9.3.2），最小二乘拟合为

$$\min_{A_x, A_y, S} \|\tilde{X} - [A_y \odot A_x] S^T\|_F \quad (9.3.13)$$

利用 LS 准则更新 S ：

$$\hat{S}^T = [\hat{A}_y \odot \hat{A}_x]^+ \tilde{X} \quad (9.3.14)$$

其中， \hat{A}_x 和 \hat{A}_y 分别为之前得到的 A_x 、 A_y 的估计值。根据式（9.3.5），LS 拟合为

$$\min_{A_x, A_y, S} \|\tilde{Y} - [A_x \odot S] A_y^T\|_F \quad (9.3.15)$$

\tilde{Y} 为含噪信号。利用 LS 准则更新 A_y ：

$$\hat{A}_y^T = [\hat{A}_x \odot \hat{S}]^+ \tilde{Y} \quad (9.3.16)$$

其中， \hat{A}_x 和 \hat{S} 分别为先前得到的 A_x 和 S 的估计值。类似地，根据式（9.3.6），LS 拟合为

$$\min_{A_x, A_y, S} \|\tilde{Z} - [S \odot A_y] A_x^T\|_F \quad (9.3.17)$$

其中， \tilde{Z} 为含噪信号。利用 LS 准则更新 A_x ：

$$\hat{A}_x^T = [\hat{S} \odot \hat{A}_y]^+ \tilde{Z} \quad (9.3.18)$$

其中, $\hat{\mathbf{A}}_y$ 和 $\hat{\mathbf{S}}$ 分别为先前得到的 \mathbf{A}_y 和 \mathbf{S} 的估计值。根据式 (9.3.14)、式 (9.3.16) 和式 (9.3.18), \mathbf{S} 、 \mathbf{A}_y 和 \mathbf{A}_x 最小二乘更新直到收敛。

3. 2D-DOA 估计

部分故障的传感器阵列的输出估计 $\hat{\mathbf{X}}$ 可以通过式 (9.3.12) 得到。通过上述的 PARAFAC 分解, 得到方向矩阵估计。通过 LS 准则, 可以得到角度估计值。

考虑阵列间距为半波长的阵列, 在无噪声模型下,

$$\mathbf{h} = -\text{angle}(\mathbf{a}_x(\phi_k, \theta_k)) = [0, \pi \cos \phi_k \sin \theta_k, \dots, \pi(M-1) \cos \phi_k \sin \theta_k]^T \quad (9.3.19)$$

其中 $\mathbf{a}_x(\phi_k, \theta_k)$ 为矩阵 \mathbf{A}_x 第 k 列, 估计的阵列方向向量 $\hat{\mathbf{a}}_x(\phi_k, \theta_k)$ (矩阵 $\hat{\mathbf{A}}_x$ 的第 k 列) 可通过归一化解决尺度模糊问题。因此, 可以通过式 (9.3.19) 得到 $\hat{\mathbf{h}}$ 。利用 LS 准则,

$$\mathbf{B}\mathbf{c}_1 = \hat{\mathbf{h}} \quad (9.3.20)$$

其中

$$\mathbf{c}_1 = [c_{01}, c_{11}]^T \quad (9.3.21)$$

$$\mathbf{B} = \begin{bmatrix} 1 & 0 \\ 1 & \pi \\ \vdots & \vdots \\ 1 & \pi(M-1) \end{bmatrix} \in \mathbb{C}^{M \times 2} \quad (9.3.22)$$

因此

$$\hat{\mathbf{c}}_1 = (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \hat{\mathbf{h}} \quad (9.3.23)$$

\mathbf{A}_x 和 \mathbf{A}_y 都为 Vandermonde 矩阵, 相同的方法可以用于从 $\hat{\mathbf{a}}_y(\phi_k, \theta_k)$ (矩阵 $\hat{\mathbf{A}}_y$ 的第 k 列) 中得到 $\hat{\mathbf{c}}_2$, $\hat{\mathbf{c}}_2 = [\hat{c}_{02}, \hat{c}_{12}]^T$ 。因此, 仰角和方位角的估计可以由下式得到:

$$\hat{\theta}_k = \sin^{-1} \sqrt{\hat{c}_{11}^2 + \hat{c}_{12}^2} \quad (9.3.24a)$$

$$\hat{\phi}_k = \tan^{-1}(\hat{c}_{12} / \hat{c}_{11}) \quad (9.3.24b)$$

4. 算法步骤

至此, 已经得到了部分传感器故障的均匀面阵下的基于 PARAFAC 填充的角度估计算法。该算法步骤如下。

- ① 得到部分工作故障的阵列的输出结果。
- ② 通过 OPTSPACE 算法进行填充。
- ③ 通过 PARAFAC 分解得到方向矩阵估计。
- ④ 利用最小二乘和式 (9.3.24) 进行二维 DOA 估计。

9.3.3 仿真结果

在下面的仿真中, 采用 $M=8$ 和 $N=6$ 的均匀面阵。假设有角度分别为 $(\theta_1, \phi_1) = (10^\circ, 20^\circ)$,

$(\theta_2, \phi_2) = (25^\circ, 35^\circ)$ 和 $(\theta_3, \phi_3) = (40^\circ, 50^\circ)$ 的三个非相关信源。样本数 $L=100$ ，并且 10 个传感器在随机的 50 个样本中不会提供数据信号。图 9.3.2 和图 9.3.3 描述了在 $\text{SNR} = 0\text{dB}$ 和 $\text{SNR}=10\text{dB}$ 时的角度估计结果，可以看出，算法估计准确，且随着信噪比的增加，算法性能提高。图 9.3.4 描述了 $M=8$ 、 $N=6$ 和 $L=100$ 下的算法性能对比，可以看出，所提算法比 ESPRIT 算法拥有更好的性能。

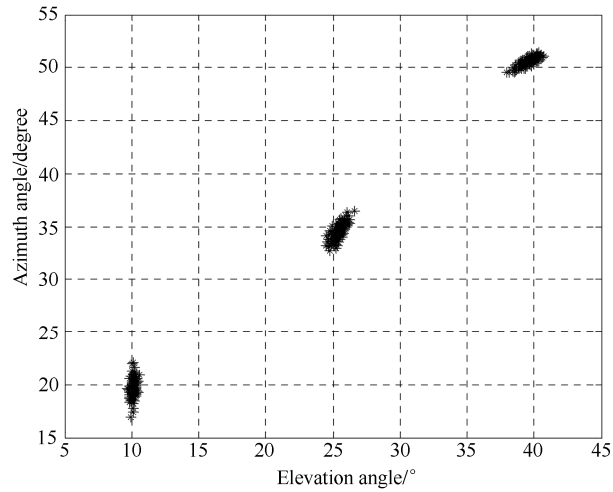


图 9.3.2 SNR=0dB 的角度估计结果

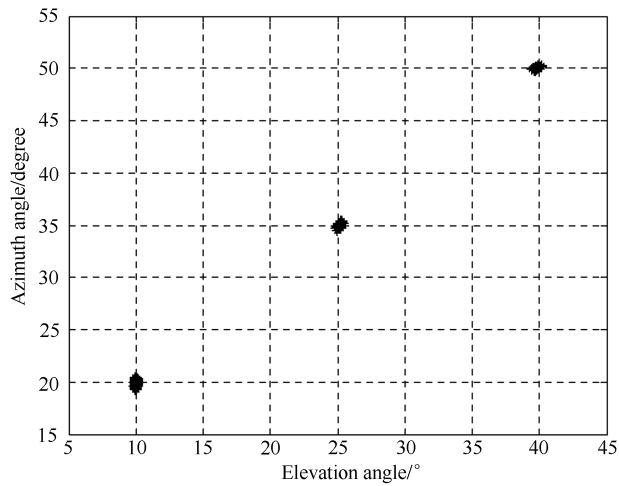


图 9.3.3 SNR=10dB 的角度估计结果

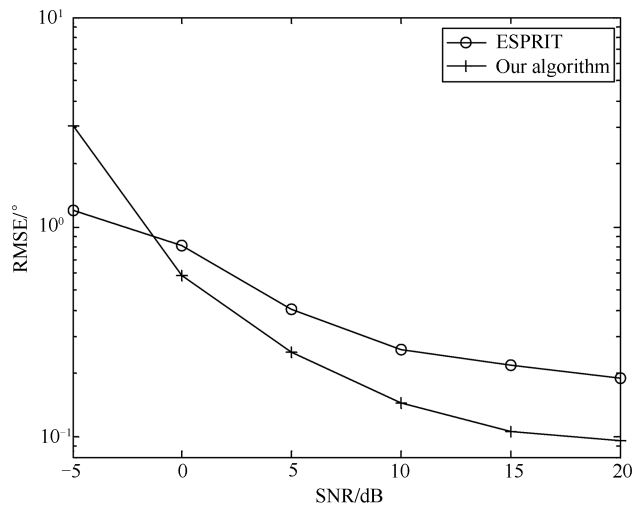


图 9.3.4 $M=8$, $N=6$ 和 $L=100$ 下的角度估计性能对比

参考文献

- [1] Xiaofei Zhang. Blind Signal Detection for Uniform Rectangular Array via Compressive Sensing Trilinear Model[J]. Advanced Materials Research (Vol.756 - 759) ,660-664.
- [2] Kruskal, J. B. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics[J]. Linear algebra and its applications, 1977, 18, (2) , pp.95-138.
- [3] Sidiropoulos, N. D. Kyrillidis, A. Multi-Way Compressed Sensing for Sparse Low-Rank Tensors[J]. IEEE Signal Processing Letters, 2012, 19 (11) :757-760.
- [4] Renzheng Cao, Xiaofei Zhang. Compressed Sensing Parallel Factor Analysis-based Joint Angle and Doppler Frequency Estimation for Monostatic MIMO Radar, IET Radar, Sonar & Navigation[J]. 2014. 8 (6) : 597 - 606
- [5] Jinli, C., Hong, G. Weimin, S. Angle estimation using ESPRIT without pairing in MIMO radar[J]. Electronics Letters, 2008, 44 (24): 1422-1423.
- [6] Zhang, X. Xu, D. Low-complexity ESPRIT-based DOA estimation for colocated MIMO radar using reduced-dimension transformation[J]. Electronics letters, 2011, 47 (4) : 283-284.
- [7] Raghunandan H. Keshavan and Sewoong Oh: OPTSPACE: A gradient Descent Algorithm on the Grassmann Manifold fo Matrix Completion. <http://arxiv.org/abs/0910.5260v2>, 2009.